

ESCUELA TÉCNICA SUPERIOR DE INGENIERÍA DE TELECOMUNICACIÓN
UNIVERSIDAD POLITÉCNICA DE CARTAGENA



Proyecto Fin de Máster

Sistema De Reconocimiento de Matrículas Basado en Visión Artificial para Control de Acceso

AUTOR: Antonio Ignacio Betancor Pérez
DIRECTORES: Cristina Vicente Chicote
Pedro J. Navarro Lorente

Octubre/ 2008



INDICE

Capítulo 1	1
OBJETIVOS Y ANTECEDENTES DEL PROYECTO.....	1
1.1. Objetivos del proyecto	1
1.2. Los primeros sistemas de parking	2
1.3. Antecedentes de la automatización	4
1.4 Estructura del proyecto	5
Capítulo 2	7
Estado de la Técnica	7
2.1. Introducción	7
2.2. Unidad de control	8
2.3. PC industrial	9
2.4. Sensores industriales	11
2.5. Control de Acceso al parking	14
2.5.1. Barreras Automáticas	14
2.5.2. Lectores de entrada/salida. Mueble de control de accesos.	16
2.6. Sistema de Visión Artificial.....	16
2.6.1. Sistema ANPR	16
2.7. Comunicación entre dispositivos	19
2.7.1. RS-485 (Comunicación serie)	19

2.7.2.	RS-232 (Comunicación serie)	20
2.7.3.	Ethernet	21
2.7.4.	Profibus (PROcess Field BUS)	21
2.7.5.	Modbus	22
2.7.6.	AS-interface	22
2.7.7.	Sistemas de visualización	24
2.8.	Base de datos	27
Capítulo 3	29
Selección de Equipo y Presupuesto	29
3.1.	Introducción	29
3.2.	Unidad de control	30
3.3.	Sistema de comunicación	33
3.4.	Sistema de control de acceso	33
3.5.	Presupuesto	34
3.5.1.	Presupuesto de la unidad de control	34
3.5.2.	Presupuesto de las vías de acceso y salida, incluido el sistema de visión.	35
3.5.3.	Presupuesto de los sensores de plazas del parking	37
3.5.5	Presupuesto de la creación de la base de datos y del sistema de reserva vía web	41
Capítulo 4	31
Solución al Problema	31
4.1.	Introducción	31
4.2.	Implementación del sistema	33
4.3.	Especificación de Requisitos	33
4.3.1.	Diagrama de transición de estados	33

4.3.2.	Ciclo de operación y modo de funcionamiento	34
4.3.3.	Tabla de descripción de estados	38
4.3.4.	Tabla de subrutinas	39
4.3.5.	Tabla de descripción de eventos	39
4.4.	Distribución del sistema	40
4.5.	Descripción del software utilizado	41
4.5.1.	Software de programación de autómatas.....	41
4.5.2.	Software de programación de la base de datos y comunicaciones con dispositivos móviles.	42
4.5.3.	Software de programación del sistema de visión.....	45
CAPÍTULO 5.....		49
SISTEMA DE RECONOCIMIENTO DE MATRÍCULAS MEDIANTE VISIÓN ARTIFICIAL		49
5.1.1.1.	Sistema de reconocimiento automático de matrículas (ANPR).....	51
5.1.1.2.	Unidad de Control	52
5.1.1.3.	Unidad de Inspección	52
5.1.1.4.	La tarjeta Matrox Meteor II	53
5.1.1.5.	La cámara digital AXIS 211	55
5.1.1.6.	Unidad de Procesado	58
5.1.1.7.	Las librerías MIL	58
5.1.1.8.	Desarrollo de la aplicación de Visión Artificial	59
5.1.1.9.	Captura de la imagen digital de la matrícula	60
5.1.1.10.	Pre-procesamiento de la imagen.....	62
5.1.1.11.	Análisis de la imagen	63
5.1.1.12.	Análisis de los resultados	65
CAPÍTULO 6.....		75
CONCLUSIONES Y FUTUROS TRABAJOS.....		75

Anexo 1	ESPECIFICACIONES TÉCNICAS	76
1.1.	Características del detector de vehículos por espiras	77
1.2.	Características del detector inductivo de vehículos	78
1.3.	Características sensor óptico reflectivo.....	79
1.4.	Características de las barreras infrarrojas	80
1.5.	Características de los sensores ultrasónicos.....	81
1.6.	Barrera vehículos RAP6.....	86
1.7.	Barrera vehículos electromecánica GARD 4	87
1.8.	Anexo Barrera vehículos RAP-.....	88
1.9.	Barrera rápida ACGRM1.....	89
1.10.	Anexo barrera BAR-XX masterasp.....	90
1.11.	Pilona automática Dakota 500.....	92
1.12.	Barreras Automáticas CT70-850-RF50-1000	93
1.13.	Barrera Automática TF-1203	97
1.14.	Sic-Transcore	100
1.15.	Sistemas de visión artificial.....	103
1.16.	Cámara fotográfica para ANPR.....	106
1.17.	SmartLPR Access	108
1.18.	Mueble de control para lectura de matrículas PNA- C4	112
1.19.	Tarjetas para captura de vídeo.....	114
	PROGRAMACIÓN EN CY C++ DE LA APLICACIÓN.....	119
2.1.	Programación en C.	120
2.2.	Programación en C++	128
BIBLIOGRAFÍA		

CAPÍTULO 1

OBJETIVOS Y ANTECEDENTES DEL PROYECTO

En este primer capítulo se presentan los objetivos del presente PROYECTO FIN de MASTER, en relación con la construcción de un sistema automatizado para el control de accesos de un aparcamiento mediante reconocimiento de matrículas y bases de datos. Asimismo, se realiza un breve repaso sobre la evolución de los aparcamientos automatizados, se enumeran las fases del Proyecto y se presenta la estructura general de la memoria

1.1. Objetivos del proyecto

El caso de estudio propuesto para este Proyecto Final de Máster consiste en el desarrollo de un sistema de control y monitorización de un parking semi-privado, con control de acceso mediante reconocimiento de matrículas y acceso a bases de datos.

El parking posee una capacidad de 200 vehículos distribuidos en dos plantas. La primera planta, a la que se accede mediante una barrera situada a pie de calle, es de acceso público y está reservada a vehículos que permanecen pequeños espacios de tiempo (menos de 24h). A la segunda planta, destinada a vehículos que tienen la plaza de aparcamiento en propiedad, se accede mediante una barrera situada dentro del parking en la primera planta.

El vehículo se considera autenticado si posee ticket (expedido por la máquina situada junto a la barrera de entrada al parking), o si su matrícula se encuentra registrada en la base de datos de vehículos de propietarios. Una plaza privada puede tener adscritas un máximo de cuatro matrículas diferentes (propietarios con más de un vehículo), pero una vez que la plaza haya sido

ocupada por cualquiera de los vehículos autorizados para ello, el resto de matrículas se considerarán no autorizadas y no se les permitirá el acceso al parking.

La salida del parking se realiza mediante la apertura de un sistema de barrera que dará acceso al casco urbano. El sistema abrirá la barrera si el vehículo es identificado como privado, mediante el sistema reconocimiento de matrículas, o si el conductor del vehículo introduce el ticket de entrada, correctamente validado por la máquina de pago de tickets. En caso contrario se activará una alarma que avisará al supervisor del parking.

El parking posee dos sistemas que suministran información del estado de las plazas de la zona de parking pública. El primero de ellos está situado en el interior y muestra, mediante un sinóptico, la localización de las plazas libres (LED-verde) y ocupadas (LED-rojo), además del número total de plazas ocupadas y libres. El sistema de información externo suministra información del número de plazas ocupadas y libres a estamentos externos como ayuntamiento, control de tráfico, vehículos entrantes, etc. Esta información será mostrada mediante paneles y testigos luminosos situados en diferentes puntos del caso urbano. Además, el sistema de información externa debe enviar al supervisor de la zona, un SMS diario con la información siguiente: fecha, hora, número de coches diarios, número de incidencias y tipo de éstas.

Ambos sistemas deberán mostrar la condición de "parking fuera servicio" en el caso de que el parking esté realizando tareas de mantenimiento, limpieza o reparación de averías. El parking cerrará el acceso a la parte pública mediante una reja protectora todos los días a las 23:00 horas (excepto los días festivos) y volverá a abrirse a las 7:30 horas.

1.2. Los primeros sistemas de parking

Los Sistemas de Estacionamiento no siempre fueron como son conocidos en la actualidad. En 1920, un joven de 22 años llamado Jack Hazard se trasladó a Los Ángeles, California, en busca de una oportunidad. Al poco tiempo y tras montar un puesto de cigarrillos, centró su atención en la adquisición de una gasolinera.

En 1922, Hazard trabó amistad con un estudiante de Instituto llamado Phil Cullen, que trabajaba en una fábrica vecina. Él necesitaba ayuda adicional y encontró en este joven un candidato para el puesto de asistente de gasolinera. Hazard le ofreció 15 dólares semanales, a Phil Cullen le

gustó esta oferta y posteriormente crearon una sociedad que duró a lo largo de los años. Y así comenzó la historia de los Sistemas de Estacionamiento.

La idea de construir aparcamientos para el personal que trabaja en las fábricas empieza a tomar forma en ciudades como Los Ángeles. Hazard y Cullen se hicieron propietarios de seis



The first System Parking Garage, 1930

gasolineras y decidieron ofrecer también servicios de aparcamiento (Figura 1.1). La idea tuvo gran éxito, pero la depresión dio un gran golpe a Hazard y Cullen. Con el fin de sobrevivir, decidieron vender radios en un intento de aumentar sus ingresos. Esto resultó ser una buena idea ya que la radio fue considerada un lujo asequible que mantenía a la gente informada durante este período de tiempo.

Figura 1.1. Primer Parking

Durante 1930, Jack Hazard aplicó su instinto empresarial en otros lugares. Phil Cullen, con 33 años, fue promocionado al cargo de Presidente. Pero el inicio de la Segunda Guerra Mundial estaba a la vuelta de la esquina. En 1942, Phil Cullen se alistó en el ejército y el Sistema de Estacionamiento estuvo inactivo hasta su regreso en 1946 [1].

A finales de los años 40's, acompañado del auge de la posguerra, el Sistema de Estacionamiento fue alcanzando nuevas alturas (Figura 1.2).

En los años 50's el Sistema de Estacionamiento estuvo presente en 30 ciudades con más de 300 localidades.

Y así, fundada en los ideales de la integridad, profesionalismo, honestidad y dedicación, el Sistema de Estacionamiento ha ido alcanzando los retos del nuevo milenio.



Stacked cars on a System Parking surface lot circa early 1950's

Figura 1.2. Aparcamiento en los 50'0

En España, el primer Sistema de Estacionamiento fue construido en Madrid (Figura 1.3) a finales de los años cuarenta. No es que por entonces hubiera en la ciudad muchos vehículos pero a los vecinos de entonces les parecían demasiados. La respuesta municipal fue construir los primeros pasos subterráneos para peatones, estrechar las aceras para aumentar las zonas de

estacionamiento y plantear reformas urbanas con el fin de mejorar la circulación. Las medidas fueron, sin embargo, insuficientes ante el crecimiento imparable del parque automovilístico por lo que el alcalde, José Finat y Escrivá de Romaní, conde de Mayalde decidió autorizar, en 1955, la construcción de un aparcamiento de 300 plazas sobre unos jardines que se encontraban semi-



abandonados, en lo que había sido solar de uno de los grandes conventos madrileños, el de santo Domingo el Real. La inauguración de este aparcamiento el 22 de diciembre de 1959, con planos del ingeniero José Ynzenga y del arquitecto Eugenio Gutiérrez Santos, fue todo un hito [2].

Figura 1.3: Primer parking de España

1.3. Antecedentes de la automatización

En 1938, un hombre llamado Claude E. Shannon, desarrolla el primer análisis simbólico de las propiedades de los circuitos de conmutación utilizando el álgebra de Boole. En 1940 se empezaron a estudiar los automatismos provistos de una cierta capacidad de memoria y los sistemas secuenciales, donde el primer método formal orientado a la síntesis de sistemas secuenciales se debe a Fuman. En 1947, Delmar S. Halder, de la compañía automovilística Ford en Detroit, se refiere por primera vez al término automatización, definiéndolo como un concepto global que debería abarcar todos los diseños y dispositivos realizados para conseguir una plena automatización de la producción. Más tarde, en 1959, C. Y. Lee desarrolla los primeros programas de decisiones binarias para la descripción de automatismos combinados. Para ello se hizo uso de herramientas como las redes de Petri, que facilitarían el diseño, el análisis y el desarrollo de automatismos secuenciales y concurrentes.

Los autómatas programables aparecen por primera vez en la industria en 1962, cuando Bedford Associates propuso, al fabricante de automóviles General Motors, un sistema de control denominado Control Digital Modular. A partir de 1973 comienzan a integrarse las funciones de comunicación en los autómatas. El Modbus, de Modicon, fue el primer bus de comunicaciones que consiguió que los PLCs pudieran comunicarse e intercambiar información con otros PLCs.

En 1980 se produjo un intento de estandarización de las comunicaciones con el protocolo MAP de General Motors. También en esta época, se redujo el tamaño de los PLCs y se paso a

programarlos con lenguajes de programación simbólicos a través de ordenadores personales (en lugar de los clásicos terminales de programación).

A partir del año 2000 se investigan y desarrollan métodos de minimización para la realización de los autómatas combinatorios y secuenciales. El desarrollo tecnológico de las últimas décadas ha hecho que la minimización de estos dispositivos vaya perdiendo interés, en favor de un aumento de la modularidad que facilite la modificación, el análisis y la puesta en marcha de estos dispositivos. [3]

1.4 Estructura del proyecto

Este Proyecto Fin de Master se ha realizado en dos etapas: (1) una primera, llevada a cabo en grupo con otros compañeros del Máster en Informática Industrial, en la que se ha recopilado información general para el diseño y la construcción del sistema, y (2) una segunda, realizada de manera individual en la que se ha desarrollado el sistema de reconocimiento de matrículas basado en técnicas de Visión Artificial.

La primera parte del Proyecto queda recogida en los siguientes capítulos:

Capítulo 1: Objetivos y antecedentes del proyecto

- ❖ Descripción de los objetivos del proyecto.
- ❖ Búsqueda de información de los primeros sistemas de parking.
- ❖ Antecedentes de la automatización.
- ❖ Estructura del proyecto.

Capítulo 2: Estado de la técnica

- ❖ Estudio del estado de la técnica en relación con los sistemas de parking.
- ❖ Acopio de información útil sobre sistemas similares actualmente en funcionamiento.

Capítulo 3: Selección del equipo y presupuesto del proyecto

- ❖ Selección de la unidad de control.
- ❖ Selección de mecanismos de acceso al parking.
- ❖ Selección del tipo de enlace de comunicación.
- ❖ Selección de la base de datos.
- ❖ Elaboración de un presupuesto para cada uno de los apartados anteriores.

Capítulo 4: Solución al problema

- ❖ Implementación del sistema.
- ❖ Especificación de requisitos.
- ❖ Ciclos de operación.
- ❖ Descripción de los estados del sistema.
- ❖ Distribución del sistema.
- ❖ Presentación y descripción del software utilizado.

A continuación, se enumeran los distintos trabajos individuales llevados a cabo por algunos de los alumnos del Máster en Informática Industrial para completar el Proyecto:

- ❖ Programación del sistema de control y monitorización del parking.
- ❖ Programación de la base de datos del parking.
- ❖ Programación del sistema de visión para el reconocimiento de matrículas.
- ❖ Integración del sistema de control, sistema de visión y base de datos.

Esta memoria sólo recoge los capítulos correspondientes al trabajo común realizado en grupo y al desarrollo de la aplicación de visión artificial para el reconocimiento de matrículas.

CAPÍTULO 2

Estado de la Técnica

En este capítulo se realiza un estudio de los diferentes dispositivos que componen un sistema de control automático de acceso a un parking como el que se pretende desarrollar y se realiza una comparativa de las ofertas existentes en el mercado para seleccionar la configuración que mejor se adecua a las necesidades del Proyecto.

2.1. Introducción

Para la realización de éste capítulo se van a evaluar las distintas opciones que ofrece el mercado con el fin de seleccionar el equipo que mejor se adecue al Proyecto en prestaciones y presupuesto. El criterio a seguir consiste en desglosar el sistema en sus componentes, analizando, para cada uno de ellos, las distintas posibilidades que ofrecen los fabricantes del sector. A continuación se puede observar un esquema funcional de un sistema automatizado general representado por la Figura 2.1.

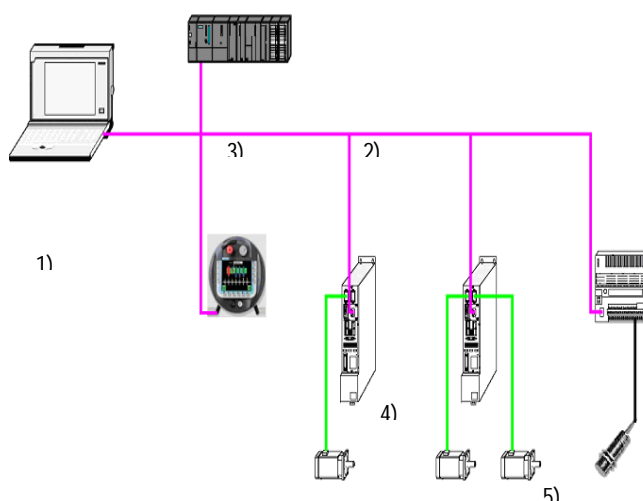


Figura 2.1. Esquema funcional.

En el esquema anterior se pueden distinguir las siguientes partes del sistema:

1. Unidad de gestión y supervisión.
2. Comunicación entre componentes del sistema.
3. Unidad Lógica de Control (MTU).
4. Unidades Lógicas Remotas (RTU).
5. Sensores y actuadores.

2.2. Unidad de control

La Unidad de control es la encargada de activar o desactivar los distintos componentes del sistema, en función de la instrucción que ejecute el microprocesador en cada momento. Así, la unidad de control (UC) interpreta y ejecuta las instrucciones almacenadas en la memoria principal y genera las señales de control necesarias para ejecutarlas [4].

Dada la cantidad de entradas y salidas que tiene que manejar el sistema, no resulta aconsejable elegir un microcontrolador como unidad de control. El uso de un PLC resulta mucho más adecuado para controlar el sistema, sobre todo gracias a la posibilidad de ampliación de los módulos de entrada/salida que ofrecen estos dispositivos

A continuación, se describen brevemente los distintos elementos que componen la unidad de control:

CPU (Central Processing Unit): Es el cerebro del PLC. Consta de uno o varios microprocesadores (según fabricante) que se programan mediante un software propio. La mayoría de ellos ofrecen varias formas de programación (lenguaje de contactos, lenguaje nemónico o instrucciones, lenguaje de funciones, Grafcet, etc.). Trabajan utilizando lógica binaria, esto es, dos estados para cada bit. Los distintos modelos de CPU ofrecen más o menos capacidad de memoria. Esto generalmente va unido a un aumento de la velocidad del reloj del procesador y a las prestaciones de cálculo o funciones matemáticas especiales que ofrecen. Hoy en día, la potencia de cálculo de los PLCs es considerable, sobre todo si se trabaja con números reales en coma flotante, proporcionando una muy buena resolución.

Bastidor o rack: El rack principal es el elemento sobre el que se encuentran situados la fuente de alimentación, su batería, la CPU, las tarjetas de entrada/salida, la memoria

EEPROM, el conmutador RUN_STOP, la tarjeta de comunicaciones, etc. Puede alojar distinto número de elementos (dependiendo del fabricante) y conectarse a otros racks similares (llamados racks de expansión) mediante un cable al efecto.

Fuente de alimentación: Es la encargada de suministrar la tensión y la corriente necesarias tanto a la CPU como a las tarjetas (según fabricante). La tensión de entrada es normalmente de 110/220VAC de entrada y 24 DCV de salida (tensión con la que normalmente se alimenta la CPU).

Memoria: La memoria del PLC se divide en dos partes: la memoria de programa y la de datos. En la parte baja de la memoria se encuentran los archivos de datos y cuando éstos terminan, empieza la zona de archivos de programa.

Módulos de entrada/salida: Se conectan al rack y se comunican con la CPU. En el caso de las entradas digitales transmiten los estados 0 o 1 del proceso (presostatos, finales de carrera, detectores, conmutadores, etc.) a la CPU. En el caso de las salidas, la CPU determina su estado tras la ejecución del programa y las activa o desactiva en consecuencia. En el caso de las entradas analógicas, estas tarjetas leen el valor analógico e internamente lo convierten en un valor digital para su procesamiento en la CPU. Esta conversión la realizan los convertidores analógico-digitales internos de las tarjetas. Normalmente se utilizan tarjetas de entradas y salidas de 24 DCV, aunque también las hay de 110 y 220 VAC, depende de las preferencias y normativas locales.

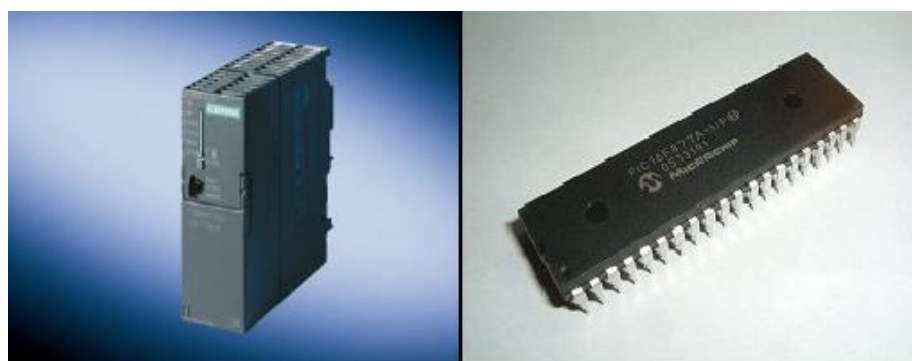


Figura 2. 2. PLC Siemens 3152DP y Microcontrolador PIC28xx

2.3. *PC industrial*

El uso de PCs (Personal Computers) en la industria, a pesar de sus grandes prestaciones (manejo de gran cantidad de datos, sistema estándar de manejo, mayores facilidades de

comunicación con otros sistemas, etc.), suele desaconsejarse debido a su poca robustez: tienen un sistema operativo generalmente inestable, pueden darse fallos en la alimentación y son poco inmunes al ruido, las vibraciones, o los golpes. Así, se hace necesario el uso de un PC mejorado o adaptado a un entorno industrial: un PC industrial.

Los PCs industriales ofrecen una mayor solidez mecánica y eléctrica, una mejor modularidad, una mayor capacidad para soportar software específico con respuestas en tiempo real y preparados para responder ante aplicaciones de tiempo crítico, proporcionan mejores protecciones ante ambientes hostiles con alto ruido electromagnético y condiciones ambientales duras.

Es posible encontrar distintos tipos de PCs industriales:

- PCs de altas prestaciones: hardware y software personalizado.
- PC en Rack: para montaje en armario, fácilmente extensible (ver Figura 2.3)



Figura 2.3. PC en Rack.

- PC de panel: integra un PC y un monitor plano de forma que ocupen el menor espacio posible, cumpliendo todos los requisitos de las MMI industriales. Por su forma compacta son ideales para sistemas móviles. En algunos casos también incluyen un teclado de membrana o pantalla táctil como se observa en la Figura 2.4.



Figura 2.4. PC de Panel.

La ventaja más obvia de una pantalla táctil es lo simple de su funcionamiento: la opción que se pulsa en la pantalla es la que se ejecuta. Esto simplifica la interfaz de hombre/máquina.

Las pantallas táctiles proporcionan un acceso rápido: no es necesario ubicar en la pantalla el puntero del ratón y, debido a que el control del tacto es más directo, los usuarios pueden trabajar más rápidamente.

Una interfaz de pantalla táctil reduce las opciones del control de un usuario. Aunque esto en un inicio pudiera parecer una desventaja, en algunas aplicaciones es beneficioso ya que aumenta la seguridad del sistema: sólo se puede acceder a las opciones de menú que ofrece la pantalla.

Las pantallas táctiles son más confiables, durables, y seguras que otros dispositivos de interfaz, como el teclado o el ratón.

- PC en caja: de pequeñas dimensiones, instalación en cabinas, pupitres y armarios. Ofrecen una funcionalidad muy básica, bajo consumo y poca disipación de calor (ver Figura 2.5).



Figura 2.5. PC en Caja.

2.4. Sensores industriales

Un sensor es un dispositivo capaz de transformar magnitudes físicas o químicas, llamadas variables de instrumentación, en magnitudes eléctricas. Las variables de instrumentación dependen del tipo de sensor y pueden ser, por ejemplo, temperatura, intensidad luminosa, distancia, aceleración, inclinación, desplazamiento, presión, fuerza, torsión, humedad, pH, etc. Puede decirse también que los sensores son dispositivos que aprovechan una de sus

propiedades con el fin de adaptar la señal que mide para que la pueda interpretar otro dispositivo.

Entre los diferentes tipos de sensores que se pueden encontrar en el mercado, tendremos sólo en cuenta aquellos que se adecuan a las necesidades del sistema [5]. En primer lugar se puede optar por un detector de vehículos por espiras, representado en la Figura 2.6, y cuyo funcionamiento consiste en una espira oscilante enterrada en el pavimento, cuya frecuencia es modificada mediante el cambio detectado por el circuito lógico que genera la señal de salida en relé. Dicho relé puede estar compuesto por una o dos espiras dependiendo de la necesidad de indicar el sentido de avance del vehículo [6].



Figura 2.6. Detector de vehículos por espiras.

Como alternativa al detector de vehículos por espiras, existe un sensor que detecta la presencia de vehículos por su masa metálica, midiendo la inductancia de una espira detectora colocada en una estructura vertical a un lado de la calzada (ver Figura 2.7). La frecuencia y la sensibilidad son testeadas por el usuario mediante jumpers ubicados en el frente del equipo. La espira es parte de un circuito oscilador que genera un campo magnético. Al atravesar el vehículo este campo, se produce un cambio en la frecuencia del oscilador. Esta variación es detectada por un circuito con microprocesador que determina, en base a la configuración del equipo, si debe producir o no la señal de salida [7].



Figura 2.7. Detector de presencia de vehículos por su masa metálica.

Como tercera alternativa, es posible optar por un sensor óptico reflectivo (ver Figura 2.8) cuyo principio de funcionamiento se basa en la emisión de un haz infrarrojo que es devuelto por un reflector y medido de forma continua por el sensor cuando no hay obstáculos. En caso de existir un obstáculo, el sensor deja de recibir el haz de luz y puede informar de la existencia de un obstáculo [6].

Como cuarta opción, se puede optar por una barrera infrarroja (ver Figura 2.9) que, como en el caso anterior, también utiliza un haz infrarrojo. En este caso, que cuando se interrumpe el haz de luz, se abre un relé que indica la detección del paso de un vehículo [6].



Figura 2. 8. Sensor óptico reflectivo.



Figura 2. 9. Barrera infrarroja

Como última opción, puede optarse por el uso de sensores ultrasónicos (ver Figura 2.10). Éstos funcionan emitiendo y recibiendo ondas de sonido a altas frecuencias. La detección del vehículo se produce cuando el sensor recibe la onda devuelta por el vehículo [8].



Figura 2. 10. Sensor ultrasónico.

2.5. Control de Acceso al parking

2.5.1. Barreras Automáticas

Dispositivo para el control de acceso de vehículos al parking. Las barreras automáticas se caracterizan porque sus maniobras de subida y bajada están supeditadas a un control eléctrico o regulación automática. La activación de la barrera puede darse por identificación del conductor del vehículo, por presencia de vehículo, etc.

Una barrera automática suele estar compuesta por:

- 1) Cuerpo barrera: sistema electromecánico o hidráulico para la elevación del sistema barrera.
- 2) Asta o mástil: Elemento barrera (que impide el paso) propiamente dicho.

Como primera opción, es posible encontrar barreras automáticas electromecánicas sin articulación (ver Figura 2.11), cuya característica principal es que los movimientos de subida y bajada están supeditados a un control eléctrico o regulación automática [9]. En segundo lugar, existe la posibilidad de contar con una barrera automática con articulación (ver Figura 2.12), cuyo funcionamiento es igual que en el caso anterior, con la única diferencia de que, al tener un brazo articulado, sólo es posible instalarla en espacios más reducidos, si bien permite un uso mucho más flexible [10]



Figura 2. 11. Barrera automática sin articulación.

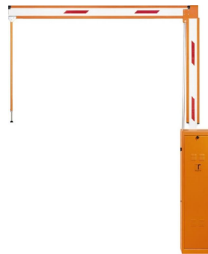


Figura 2. 12. Barrera automática con articulación.



La siguiente posibilidad es la utilización de pilonas automáticas (ver Figura 2.13), cuyo funcionamiento consiste en el hundimiento de dichas pilonas en el firme para permitir el acceso de los vehículos una vez identificados. El uso de pilonas es aconsejable en lugares cuya altura no permita la utilización de barreras elevadoras automáticas [11].

Figura 2. 13. Pilona automática

2.5.2. Lectores de entrada/salida. Mueble de control de accesos.

La primera posibilidad es utilizar un dispositivo para la lectura de tarjetas con sistema de reconocimiento RFID (ver Figura 2.14). Este tipo de sistemas se denominan, de manera abreviada, RFID (Radio Frequency Identification), ya que la identificación se realiza por radiofrecuencia. Su uso en el control de accesos hace que se gane en comodidad ya que no es necesario el contacto físico de la tarjeta con el lector. Éste es un sistema en el que el interrogador (dispositivo que lee los datos) tiene que poder leer muchas tarjetas diferentes, tantas como usuarios haya autorizados.



Figura 2. 14. Mueble de control de acceso y tarjeta RFID.

Otra opción, es identificar a los abonados automáticamente mediante la lectura e identificación de la matrícula de su vehículo mediante un sistema visión artificial. La imagen capturada por la cámara es procesada y contrastada con la información almacenada en la base de datos. Si la matrícula es correcta, se enviará la orden de apertura de la barrera a la unidad de control. Para los clientes ocasionales, se utilizará una máquina expendedora de tickets.

2.6. Sistema de Visión Artificial

El sistema de Visión Artificial será el encargado del reconocimiento de las matrículas de los vehículos que deseen acceder al parking. Para ello, se utilizarán técnicas de reconocimiento automático de caracteres (Optical Character Recognition, OCR).

2.6.1. Sistema ANPR

El **reconocimiento automático de matrículas** (Automatic Number Plate Recognition o ANPR, en inglés) es un método de vigilancia en masa que utiliza reconocimiento óptico de caracteres en imágenes para leer las matrículas de los vehículos (ver Figura 2.12). A partir de 2005, los

sistemas pueden escanear las matrículas con una frecuencia aproximada de una por segundo en vehículos con velocidades de hasta 160 km/h. El ANPR se puede utilizar para almacenar las imágenes capturadas por las cámaras fotográficas, así como el texto de la matrícula. En algunos casos, el ANPR se puede configurar para almacenar también una fotografía del conductor. Estos sistemas a menudo utilizan iluminación infrarroja para hacer posible que la cámara pueda tomar fotografías en cualquier momento del día. La tecnología ANPR tiende a ser específica para una región, debido a la variación entre matrículas de un lugar a otro.

El software del sistema de Visión Artificial se ejecutará sobre un PC convencional y estará enlazado con el resto de las aplicaciones o bases de datos. El sistema inicialmente utiliza una serie de técnicas de manipulación de la imagen para detectar, normalizar y realzar la imagen de la matrícula (pre-procesamiento de la imagen) para, a continuación, aplicar el reconocimiento óptico de caracteres para extraer los datos alfanuméricos de la matrícula. Esta información, convertida ahora en pequeños paquetes de datos, se puede transmitir fácilmente (de ser necesario) a otros dispositivos remoto para su almacenamiento o procesamiento posterior [12].

Hay cinco algoritmos que resulta necesario implementar para identificar una matrícula:

- 1) Localización de la matrícula - responsable de encontrar y aislar la matrícula en la imagen.
- 2) Orientación y tamaño de la matrícula - compensa los ángulos que hacen que la matrícula parezca "torcida" y ajusta las dimensiones al tamaño requerido.
- 3) Normalización - ajusta el brillo y el contraste de la imagen.
- 4) Segmentación y reconocimiento de los caracteres - encuentra los distintos caracteres presentes en la matrícula.
- 5) Análisis sintáctico y geométrico - comprueba los caracteres encontrados y sus posiciones, utilizando los patrones correspondientes a las matrículas específicas de cada país o región.



Figura 2. 15. reconocimiento automático de matrículas.

Los sistemas ANPR también pueden ser usados para:

- Control de cruce de fronteras
- Estaciones de servicios, para llevar un registro de los conductores que abandonan la estación sin realizar el pago
- Herramientas para llevar un registro de patrones de uso
- Sistemas de gestión de tráfico, para determinar el flujo de tráfico usando el tiempo que tardan los vehículos en pasar por dos sitios dotados de ANPR [11].

Además de la elección de la cámara (ver Figura 1.16), es necesario seleccionar también una tarjeta de video para el procesado de la imagen en el ordenador.



Figura 2. 16. Cámaras de Visión Artificial.

2.7. Comunicación entre dispositivos

Las redes de comunicaciones industriales son los instrumentos que permiten la integración y comunicación entre todos los componentes de un sistema industrial automatizado. Son dispositivos digitales capaces de manejar una ingente cantidad de datos con una alta precisión y pueden transmitir variables secuencialmente a alta velocidad. Estos dispositivos siguen unos protocolos estándar para el intercambio de datos. Cabe destacar entre todos los tipos de comunicación industrial los siguientes:

- ❖ Comunicación serie.
- ❖ Comunicación por bus de campo.
- ❖ Redes de Área local.

2.7.1. RS-485 (Comunicación serie)

La comunicación RS-485 permite la transmisión multipunto diferencial y es ideal para transmitir a altas velocidades sobre largas distancias y a través de canales ruidosos (35Mbps hasta 10 Metros y 100 Kbps hasta 1200 Metros). El medio físico de transmisión (ver Figura 2.17) es un par entrelazado que admite hasta 32 estaciones en un solo hilo, con longitud máxima de 1200 metros y comunicación semi-dúplex [13].

Especificaciones Técnicas:

- ❖ Interfaz diferencial.
- ❖ Conexión multipunto.
- ❖ Alimentación +5V.
- ❖ Hasta 32 estaciones.
- ❖ Velocidad máxima de 10 Mbps(a 12 metros).
- ❖ Longitud máxima de alcance 1200 metros(a 100Kbps).
- ❖ Rango de -7V a +12V.



Figura 2. 17. Medios físicos de transmisión RS-485.

2.7.2. RS-232 (Comunicación serie)

La comunicación RS-232 (ver Figura 2.18) se basa en el uso de una interfaz, que designa una norma para el intercambio serie de datos binarios entre un equipo terminal de datos (DTE) y un equipo de comunicación de datos (DCE). Está diseñada para distancias cortas, de 15 metros o menos, y para velocidades de comunicación bajas (no más de 20 Kb). Puede trabajar en distintos tipos de canal destacando simplex, half dúplex y full dúplex [14].

Voltaje	Señal	Nivel Lógico	Control
+3 a +15	Espacio	0	On
-3 a -15	Marca	1	Off

Tabla 2.1: Comunicación Serie.



Figura 2.18: Medios Físicos de transmisión serie

2.7.3. Ethernet

Se basa en una tecnología de computadoras de área local. Utiliza tramas de datos definidas por el protocolo CSMA/CD. Permite hasta 5000 metros de distancia máxima con una velocidad de transmisión de 1000 Mbps.



Figura 2.19. Medios físicos de transmisión con Ethernet.

Trama de Ethernet						
Preámbulo	SOF	Destino	Origen	Tipo	Datos	FCS
7 bytes	1 byte	6 bytes	6bytes	2 bytes	46 a 1500 bytes	4 bytes

Tabla 2.2. Características de la transmisión Ethernet.

2.7.4. Profibus (PROcess FieId BUS)

Profibus es posiblemente el bus de campo industrial más utilizado (ver Figura 2.20). Se trata de una red abierta, estándar e independiente de cualquier fabricante y se adapta muy bien a las condiciones de las aplicaciones de automatización industrial. La velocidad de transmisión va desde 9,6 hasta 12 Mbit/seg. con un número máximo de estaciones de 127 y unas distancias máximas alcanzables de 1200 metros hasta 93,75 Kbaudios. Puede tener conexiones de tipo direccionales, multicast o broadcast [15].



Figura 2. 20. Medios físicos de transmisión con Profibus.

2.7.5. Modbus

Modbus (ver Figura 2.21) se basa en un protocolo de comunicaciones modelo OSI basado en la arquitectura maestro/esclavo o cliente/servidor diseñado por Modicom para su gama de PLCs. Es el protocolo de comunicaciones que goza de mayor disponibilidad para la conexión de dispositivos electrónicos industriales ya que es público, tiene una implementación fácil con poco desarrollo y maneja bloques de datos sin restricciones [16].



Figura 2. 21. Medios físicos de transmisión con Modbus.

2.7.6. AS-interface

Es un bus estándar (internacional y europeo) de sensores y actuadores para un nivel de campo más bajo que Profibus, situándose en la parte más baja de la pirámide de control y conectando los sensores y actuadores con el maestro del nivel de campo, pudiendo ser dichos maestros autómatas o PCs situados en niveles bajos de control (ver Figura 2.22) [17].

Las características principales de AS-Interface son:

- ❖ Ideal para la interconexión de sensores y actuadores binarios.
- ❖ A través del cable AS-i se transmiten datos y alimentación.
- ❖ Cableado sencillo y económico. Se puede emplear cualquier cable bifilar de 2 x 1.5 mm² no trenzado ni apantallado.

- ❖ El cable específico para AS-i (el cable amarillo) es autocatratizante y está codificado mecánicamente para evitar su polarización incorrecta.
- ❖ Gran flexibilidad de topologías, que facilita el cableado de la instalación.
- ❖ Sistema monomaestro, con un protocolo de comunicación con los esclavos muy sencillo.
- ❖ Ciclo del bus rápido. Máximo tiempo de ciclo 5 ms con direccionamiento estándar y 10 ms con direccionamiento extendido.
- ❖ Permite la conexión de sensores y actuadores No AS-i mediante módulos activos.
- ❖ Hasta 124 sensores y 124 actuadores binarios con direccionamiento estándar.
- ❖ Hasta 248 sensores y 186 actuadores binarios con direccionamiento extendido.
- ❖ Longitud máxima de cable de 100 m uniendo todos los tramos, o hasta 300 m con repetidores.
- ❖ La revisión 2.1 del estándar facilita la conexión de sensores y actuadores analógicos.
- ❖ Transmisión por modulación de corriente que garantiza un alto grado de seguridad.
- ❖ Detección de errores en la transmisión y supervisión del correcto funcionamiento de los esclavos por parte del maestro de la red.
- ❖ Cables auxiliares para la transmisión de energía: Cable Negro (24 V DC) y Rojo (220 V AC).
- ❖ Grado de Protección IP-65/67 para ambientes exigentes.
- ❖ Cumple la normativa IP-20 para aplicaciones en cuadro.
- ❖ Temperaturas de funcionamiento entre -25°C y +85°C.

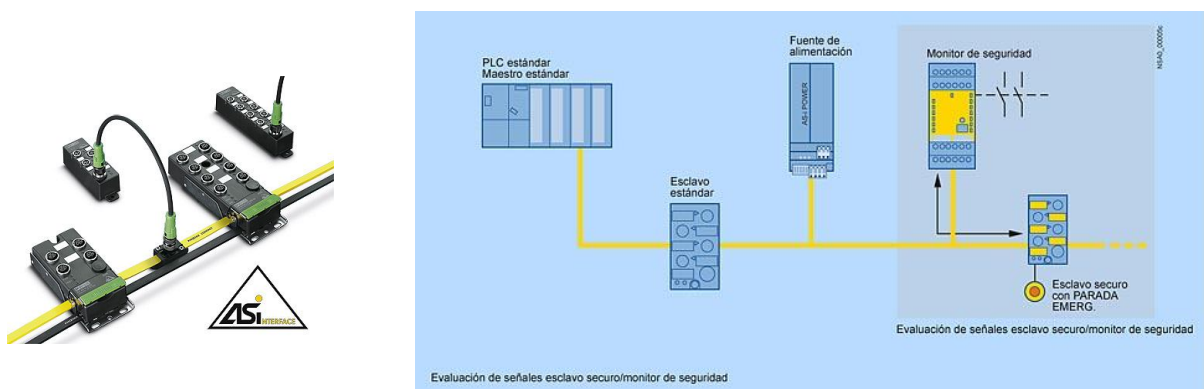


Figura 2. 22. AS-Interface.

2.7.7. Sistemas de visualización

En este punto se abordarán los sistemas de visualización y control de procesos industriales. Las diferentes opciones que se presentan en el mercado son: el uso de SCADA, HMI, paneles de operador y PCs industriales.

HMI (Human Machine Interface)

Una interfaz hombre-máquina o HMI (ver Figura 2.23) es el dispositivo que presenta los datos a un operador humano y a través del cual éste controla el proceso. La industria del HMI nació esencialmente de la necesidad de estandarizar la manera de monitorizar y controlar múltiples sistemas remotos, PLCs y otros mecanismos de control. Un HMI puede tener vínculos también con una base de datos para proporcionar las tendencias, los datos de diagnóstico y el manejo de la información así como un cronograma de procedimientos de mantenimiento, información logística, esquemas detallados para un sensor o máquina, incluso sistemas expertos con guía de resolución de problemas.



Figura 2. 23. Human-Machine Interface (HMI).

SCADA (Supervisory Control And Data Acquisition)

El termino SCADA usualmente se refiere a un sistema central que visualiza y controla un sitio completo o un sistema que se extiende sobre una gran distancia (kilómetros / millas). La mayor parte del control del sistema es, en realidad, realizada automáticamente por una Unidad Terminal Remota (RTU) o por un Controlador Lógico Programable (PLC). Las funciones de control del servidor están casi siempre restringidas a reajustes básicos del sitio o capacidades de nivel de supervisión. Por ejemplo, un sistema SCADA puede permitirle a un operador cambiar el punto de consigna (set point) de control para el flujo, y permitirá grabar y mostrar cualquier condición de alarma como la pérdida de un flujo o la detección de una temperatura excesivamente alta. Por el contrario, con un sistema de visualización y un PLC no se podría actuar sobre el proceso. La realimentación del lazo de control es cerrada a través del RTU o el PLC; el sistema SCADA visualiza el desempeño general de dicho lazo (ver Figura 2.24).

Un sistema SCADA comprende todas aquellas soluciones de aplicación relacionadas con la captura de información de un proceso o planta industrial (aunque no es absolutamente necesario que pertenezca a este ámbito), para que, con esta información, sea posible realizar una serie de análisis o estudios con los que se pueden obtener valiosos indicadores que permitan una retroalimentación sobre un operador o sobre el propio proceso [18].

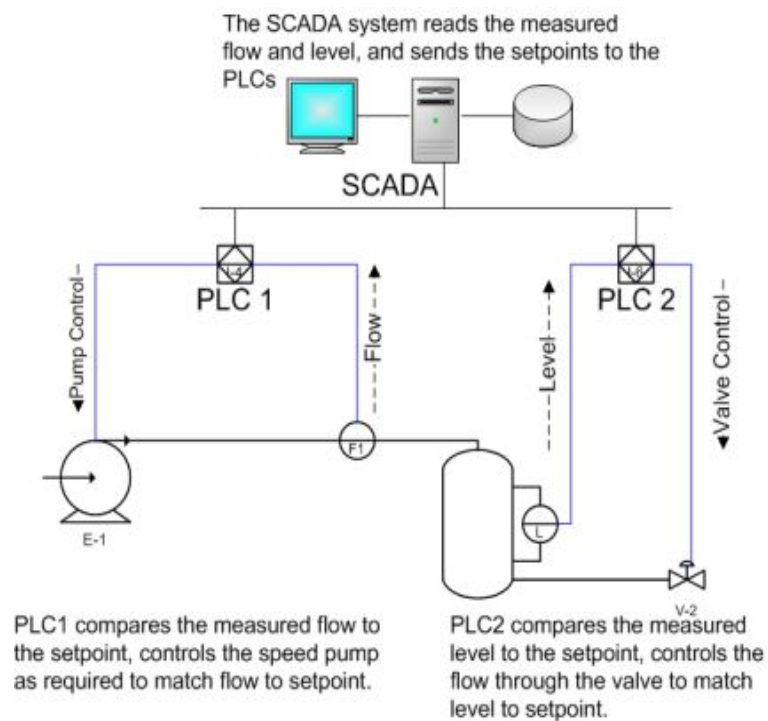


Figura 2.24. Sistema SCADA.

La Figura 2.25 es un ejemplo de la aplicación del sistema SCADA en áreas industriales.

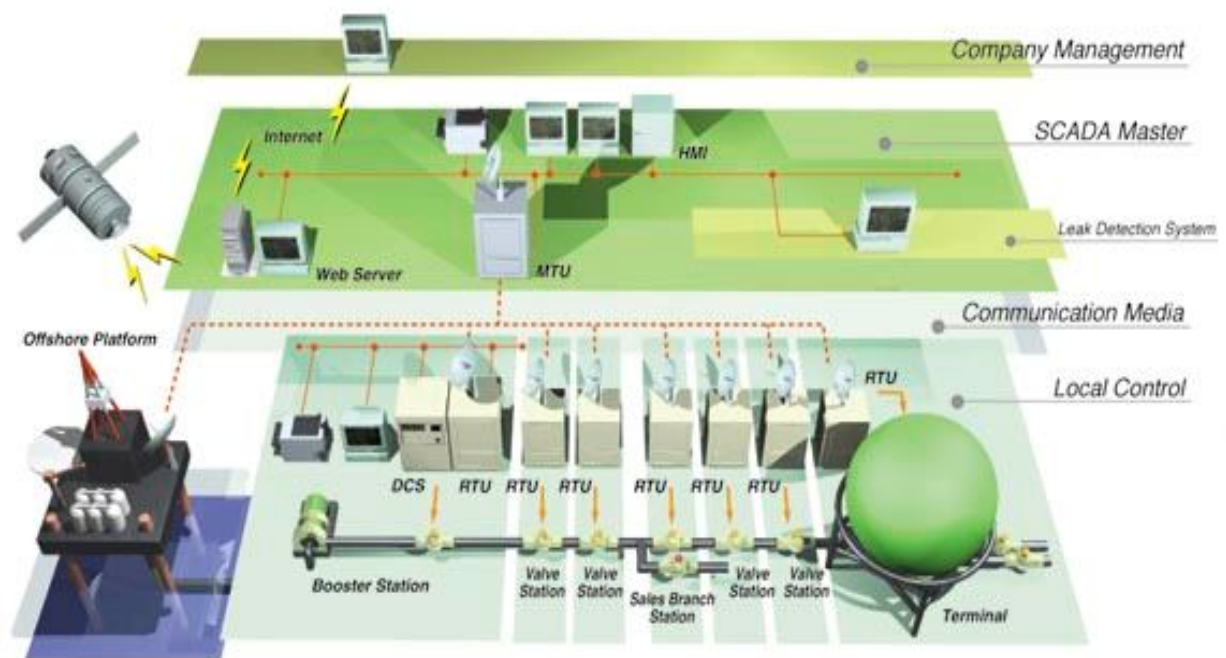


Figura 2. 25. Topología de una planta industrial automatizada con un sistema SCADA.

Los sistemas SCADA son de aplicación en los siguientes sectores industriales:

- ❖ Monitorización de procesos químicos, físicos o de transporte en sistemas de suministro de agua, para controlar la generación y distribución de energía eléctrica, de gas o en oleoductos y otros procesos de distribución.
- ❖ Gestión de la producción (facilita la programación de la fabricación).
- ❖ Mantenimiento (proporciona magnitudes de interés tales para evaluar y determinar modos de fallo, MTBF, índices de Fiabilidad, entre otros);
- ❖ Control de Calidad (proporciona, de manera automatizada, los datos necesarios para calcular índices de estabilidad de la producción CP y [CPk](#), tolerancias, índice de piezas NOK/OK, etc.).
- ❖ Administración (actualmente pueden enlazarse los datos del SCADA con un servidor ERP o Enterprise Resource Planning, o un sistema de planificación de recursos empresariales).
- ❖ Tratamiento histórico de información (mediante su incorporación en bases de datos).

A diferencia de los HMI los SCADA pueden monitorizar y también supervisar el proceso y además gestionan más información comunicándose con una base de datos. Ofrecen una arquitectura abierta y flexible, totalmente ampliable y adaptable, y una conectividad con otras

aplicaciones y bases de datos locales o distribuidos en redes de comunicación, siendo la supervisión otro punto fuerte para la observación de la evolución de las variables de control.

2.8. Base de datos

Una base de datos es un conjunto de datos pertenecientes a un mismo contexto y almacenados sistemáticamente para su posterior uso. En la actualidad y debido al desarrollo tecnológico de campos como la informática y la electrónica, la mayoría de las bases de datos están en formato digital, lo cual ofrece un amplio rango de soluciones al problema de almacenar datos. Los sistemas gestores de bases de datos (SGBD) son los encargados de permitir el almacenamiento y posterior acceso a los datos de forma rápida y estructurada [19].

Las aplicaciones más usuales de las bases de datos tienen que ver con la gestión de empresas e instituciones públicas, y en la actualidad también son ampliamente utilizadas en entornos científicos con el objeto de almacenar la información experimental.

Las bases de datos se pueden clasificar según:

1) La variabilidad de los datos almacenados:

Las llamadas bases de datos estáticas, son bases de datos de sólo lectura, utilizadas para almacenar datos históricos que posteriormente se pueden utilizar para estudiar el comportamiento de un conjunto de datos en el tiempo, proyecciones y toma de decisiones.

Las bases de datos donde la información almacenada se modifica con el tiempo, permitiendo operaciones como actualización, adición de datos y operaciones fundamentales de consulta, son las denominadas bases de datos dinámicas.

2) Contenido

Esta clasificación se realiza atendiendo a la naturaleza de los datos de la base de datos. Cabe destacar, entre otras, las bases de datos bibliográficas, las de texto completo, los directorios o "bibliotecas" de información biológica.

3) Modelos de bases de datos

Además de la clasificación por la función de las bases de datos, éstas también se pueden clasificar atendiendo a su modelo de administración de datos. Un modelo de datos es básicamente una descripción de algo conocido como "contenedor de datos" (donde se guarda la información), así como los métodos para almacenar y recuperar información de esos contenedores. Se pueden encontrar los siguientes modelos de bases de datos: bases de datos jerárquicas, de red, relacionales, multidimensionales, orientadas a objetos, documentales, deductivas y distribuidas.

CAPÍTULO 3

Selección de Equipo y Presupuesto

Para la realización del Proyecto es necesaria la elección de la tecnología que mejor se adapte a los requisitos físicos, económicos y funcionales del sistema que se pretende realizar. Por ello, en este capítulo, se van a evaluar distintos dispositivos y a seleccionar aquéllos que mejor satisfacen las condiciones establecidas. Posteriormente con los dispositivos seleccionados se realizara el presupuesto del Proyecto.

3.1. Introducción

En el capitulo anterior se ha establecido cuál es el panorama actual en relación con la tecnología que se puede utilizar en un proyecto de estas características. Una vez que se conocen cuales son los dispositivos de los que dispone el mercado, es el momento de seleccionar aquéllos que mejor se adecuen a los requisitos del proyecto. Para ello, se va a dividir este capítulo en cuatro puntos bien diferenciados que se corresponden con las distintas unidades de las que estará compuesto el sistema final. De este modo, se facilitará la elaboración del presupuesto del Proyecto.

3.2. **Unidad de control**

Una vez adoptada la solución del autómatas para implementar el algoritmo de control del parking, se plantea ahora la necesidad de seleccionar, entre la amplia oferta del mercado, el equipo más adecuado. La decisión debe basarse en un análisis sistemático, no sólo de las características actuales de la tarea de control, sino también de las necesidades futuras.

Las restricciones que debe cumplir el PLC seleccionado para la unidad de control son las siguientes:

- 120 entradas digitales.
- 20 salidas digitales.
- Comunicación con los paneles informativos (vía GSM).
- Comunicación con el sistema de visión.
- Comunicación con la base de datos.
- SCADA del sistema.

Por lo tanto una vez barajadas todas las posibilidades que se ofrecen desde SIEMENS, OMRON ó TELEMECANIQUE, entre otras, se estudia una de las opciones ofrecidas por SIEMENS. SIEMENS dispone de una gama amplia de PLCs dependiendo de las necesidades de cada sistema que queramos controlar. Entre las tres series que ofrece esta marca, S7-200, S7-300, S7-400, automáticamente se puede descartar la serie 400 ya que, aunque se trata de una CPU muy potente, su elevado precio la convierte en no viable. Por lo tanto, las alternativas que se van a valorar son las ofrecidas por las series 200 y 300 [20].

Finalmente se ha optado por la serie S7-300 ya que, aunque con un serie 200 podríamos cumplir las expectativas del sistema, la serie 300 nos da un mayor margen debido a que supera las capacidades que ofrece la serie 200, tanto en potencia como en el número de entradas que se pueden conectar. Concretamente se ha seleccionado la CPU 315-2DP. Esta unidad debe ir acompañada de una serie de tarjetas y componentes que permitan integrar esta CPU en el amplio sistema que se quiere implementar. Todos los componentes necesarios aparecerán detallados junto con su precio en el apartado de elaboración del presupuesto. A continuación, en la Figura 3.1, puede verse una relación de las características principales del PLC seleccionado para abordar la aplicación que nos ocupa [21].

Alimentación	
Tensión de entrada	
Valor nominal (DC)	24 V
Rango admisible, límite inferior (DC)	20,4 V
Rango admisible, límite superior (DC)	28,8 V
Consumo	
Intensidad de cierre, típ.	2,5 A
P _t	0,5 A ² s
Consumo (en marcha en vacío), típ.	60 mA
De la tensión de alimentación L+, máx.	800 mA
Pérdidas, típ.	2,5W
Memoria	
Tipo de memoria	
Memoria de trabajo	
Integrada	128 Kbyte
Ampliable	No
Memoria de carga	
Enchufable (MMC)	Sí
Enchufable (MMC), máx.	8 Mbyte
Conservación de datos en MMC (tras última programación), mín.	10 Year
CPU/bloques	
Nº de bloques (total)	1024
DB	
Cantidad, máx.	1023
Tamaño, máx.	16 Kbyte
FB	
Cantidad, máx.	1024
Tamaño, máx.	16 Kbyte
FC	
Cantidad, máx.	1024
Tamaño, máx.	16 Kbyte
OB	
Tamaño, máx.	16 Kbyte
CPU/tiempos de ejecución	
para operaciones de bits, mín.	0,1 µs
para operaciones de palabras, mín.	0,2 µs
para aritmética en coma fija, mín.	2 µs
para aritmética en coma flotante, mín.	3 µs
Contadores S7	
Cantidad	256

Temporizadores S7	
Cantidad	256
Área de direcciones	
Área de direcciones de periferia	
Entradas	2 Kbyte
Salidas	2 Kbyte
Imagen del proceso	
Entradas	128 Byte
Salidas	128 Byte
Canales digitales	
Entradas	16384
Salidas	16384
Entradas, de ellas centralizadas	1024
Salidas, de ellas centralizadas	1024
Canales analógicos	
Entradas	1024
Salidas	1024
Entradas, de ellas centralizadas	256
Salidas, de ellas centralizadas	256
CPU/programación	
Lenguaje de programación	
STEP 7	Sí; STEP 7 V5.2
KOP	Sí
FUP	Sí
AWL	Sí
SCL	Sí
CFC	Sí
GRAPH	Sí
HiGraph®	Sí
Dimensiones	
Ancho	40 mm
Alto	125 mm
Profundidad	130 mm
Peso	
Peso, aprox.	290g

Figura 3.1. Características principales de la CPU seleccionada (315-2DP de Siemens).

Además de los componentes necesarios para el correcto funcionamiento del sistema, también se deberá evaluar la posibilidad o necesidad de añadir más componentes al PLC dependiendo del tipo de bus de comunicaciones que se utilice, caso que se analizará en el apartado siguiente.

3.3. Sistema de comunicación

Como se ha mostrado en el capítulo 2 existe una gran variedad de estándares y modelos para implementar una red de comunicaciones industrial.

Los sistemas de comunicación planteados se pueden dividir en tres grandes grupos:

- Comunicación en Serie: RS 232, RS-485.
- Redes de Área Local: Ethernet.
- Buses de campo: Profibus, Modbus, AS-Interface

Un sistema con las características del que se desea implementar debe integrar varios tipos de comunicación, según el nivel de la pirámide en el que se esté:

- Comunicación entre PLC's (en el caso que sea necesario utilizar más de uno): se llevaría a cabo mediante un conector DB9 que se correspondería con el estándar RS-485.
- Comunicación PLC – SCADA: Ethernet
- Comunicación PLC – SENORES/ACTUADORES: Bus de campo (AS-INTERFACE).

Por lo tanto es necesario añadir módulos al PLC seleccionado para llevar a cabo la comunicación con los diversos elementos seleccionados. En el apartado en el que se presenta el presupuesto, se podrá ver desglosados los módulos necesarios para el correcto funcionamiento de todo el equipo.

3.4. Sistema de control de acceso

En este punto se van a seleccionar los sistemas de control de acceso de entre los disponibles en el mercado (barreras, expendedores, etc.), así como los sistemas que detectan la presencia de vehículos (sensores).

El encargado de la coordinación de todos estos dispositivos será el PLC, que recogerá información de estado de estos equipos y generará las pertinentes respuestas. El sistema de control de acceso deberá detectar, a la entrada del parking, la presencia de un coche con el propósito de activar la cámara y la salida del ticket del expendedor en caso de ser necesario.

Una vez analizadas todas las posibilidades que ofrece el mercado (ver Anexo 1), así como el coste y rendimiento de cada una de las opciones barajadas, se ha optado usar unas barreras de

tres metros de brazo colocadas: dos a la entrada del parking una general y otra exclusivamente para residentes y una más a la salida del mismo. Así mismo, se ha optado también por la inclusión de un mueble para la entrada y otro para la salida, ambos con una cámara de visión artificial integrada así como un iluminador.

3.5. Presupuesto

3.5.1. Presupuesto de la unidad de control.

A continuación se muestra el presupuesto ofrecido por la empresa SIEMENS S.A. para el PLC seleccionado [22].

Pos.	Cant.	Artículo	Importe Unid.	Importe Total
1		S7-300: 128 entradas / 128 salidas digitales		
1.1	1	SIMATIC S7-300, perfil soporte longitud =530mm 6ES7390-1AF30-0AA0	45,26	45,26
1.2	1	SIMATIC S7-300, fuente de alimentación PS 307, 120/230 V AC; 24 V DC, 10 A 6ES7307-1KA01-0AA0	208,32	208,32
1.3	1	SIMATIC S7-300, CPU 315-2DP CPU con MPI, fuente de alimentación integrada DC 24V, memoria central de 128 KBYTES, 2 puertos DP-MASTER/SLAVE, requiere MICRO MEMORY CARD 6ES7315-2AG10-0AB0	2.025,72	2.025,72
1.4	1	SIMATIC S7, Micro Memory Card para S7-300/C7/S7-200 IM 151 CPU, 3,3 V NFLASCH, 128 Kbyte 6ES7953-8LG11-0AA0	102,65	102,65
1.5	4	SIMATIC S7-300, tarjeta de entradas digitales SM 321, con separación galvánica 32 ED, 24 V DC (1 x 32 ED). 40 polos 6ES7321-1BL00-0AA0	423,36	1.693,44
1.6	4	SIMATIC S7-300, tarjeta de salidas digitales SM 322, con separación galvánica, 32 SD, 24 V DC, 0,5 A corriente total 8 A 6ES7322-1BL00-0AA0	586,53	2.346,12
1.7	8	SIMATIC S7-300, conector frontal 392 con terminales de tornillo, 40 polos 6ES7392-1AM00-0AA0	46,53	372,24

Total Partida (Euros) 6.793,75

Pesetas 1.130.385

Total Oferta (Euros) 6.793,75

Pesetas 1.130.385

Todos los importes, salvo los indicados expresamente, están en la moneda de emisión de la Oferta (Euros)

3.5.2. Presupuesto de las vías de acceso y salida, incluido el sistema de visión.



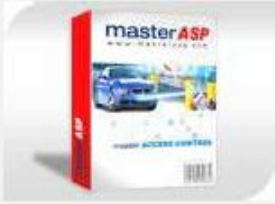
A continuación se especifica el precio de los sistemas de acceso, salida y reconocimiento de matrículas del sistema [23].

Salida		Cant	Prec.Unit	Importe
	Barrera masterASP de 3m. menos de 2 segundos - Funcionamiento Continuo - Fotocélula Seguridad Personas - Bucle Seguridad Vehículos - Botonera en Centro Control	1	2.100,00 €	2.100,00 €
	Mueble soporte de Cámaras y Electrónica - Cámara de video - Iluminador - Electrónica de control - 2 Bucle de presencia de vehículo	1	1.950,00 €	1.950,00 €
Subtotal				4.050,00 €

Presupuesto Sistema de Control de Accesos

Lectura de matrículas en parking de vehículos. Dispone de dos entradas y una salida, controladas mediante barreras de apertura rápida

Entrada		Cant	Prec.Unit	Importe
	Barrera masterASP de 3m. menos de 2 segundos - Funcionamiento Continuo - Fotocélula Seguridad Personas - Bucle Seguridad Vehículos - Botonera en Centro Control	2	2.100,00 €	4.200,00 €
	Mueble soporte de Cámaras y Electrónica - Cámara de video - Iluminador - Electrónica de control - 2 Bucle de presencia de vehículo	2	1.950,00 €	3.900,00 €
Subtotal				8.100,00 €

Centro de Control		Cant	Prec.Unit	Importe
	Caja de Control Central - Convertidor CAN a RS232 - Fuente de Alimentación - Armario 300 x 300 mm	1	900,00 €	900,00 €
	Servidor de Barreras - PC HP-EVO P IV, 3,3 MHz, 1GB, 120GB - Monitor TFT 19"	1	995,00 €	995,00 €
	Tarjeta Capturadora de Video Tarjeta Grabber Card 4 puertos 70 imagenes/seg	1	520,00 €	520,00 €
	Licencia de Software Master Access Control	1	5.500,00 €	5.500,00 €
			Subtotal	7.915,00 €

Instalación, Formación, Puesta en Marcha y Garantía		Cant	Prec.Unit	Importe
	Instalación Incluye: - Instalación de los Equipos - Instalación del Software - Transportes y Desplazamientos No Incluye: - Obra Civil - Conexionado al Cuadro 220 V - Cableados superiores a 50m.	1	2.000,00 €	2.000,00 €
	Formación y Puesta en Marcha - Parametrización Inicial - Carga de Datos Inicial - Formación Usuarios - Desplazamientos	1	1.000,00 €	1.000,00 €
	Garantía 1 año Incluye: - Recambios Y Transportes - Mano de Obra - Servicio Hot Line No Incluye: - Uso Inadecuado del Material - Accidentes	1	0,00 €	0,00 €
Subtotal			3.000,00 €	
Total Oferta: 23.065,00 €				

3.5.3. Presupuesto de los sensores de plazas del parking

El equipo a presupuestar para esta parte del parking estará formado por:

- 150 sensores de presencia.
- 3 amplificadores.
- 1 master general.
- Software.
- Instalación y montaje.

Los precios ofrecidos por una empresa del sector son los que se muestran a continuación [24].

DESCRIPCIÓN DEL EQUIPO		
<u>SENSORES POSICIONADORES CUML-1</u>		
Sensores electrónicos por ultrasonidos, con indicador de estado en posición de reposo, piloto indicador de función, reset de regulación para ajuste de sensibilidad en altura de techo. Salida para indicador exterior de estado. Carcasa de seguridad y estanco. Alimentación a 24 VC, con bajo consumo. Micro Procesador de comunicación. Puerto de comunicación RS-485		
150 UDES	85,00	12.750,00
<u>AMPLIFICADOR CONTROLADOR DE ZONA Z64</u>		
Elemento electrónico de interconexión de sensores, modulo compuesto de: Fuente de alimentación 24 VC 2 A Micro Procesador de movimientos Display informativo secundario Regenerador de señal 485 Comunicación para 64 terminales. Cuadro de seguridad y estanco Alimentación 220 VA.		
3 UDES	94,00	282,00
<u>CUADRO MASTER GENERAL CM-1</u>		
Elemento Electrónico de interconexión de módulos Master compuesto de: Comunicación con PC RS-232 Micro Procesador de movimientos Display LCD alta resolución Puerto de comunicación RS-485 Pulsadores de Ajuste de plazas y zonas Pulsadores de reset Cuadro de seguridad y estanco Alimentación 220 VA.		
1 UDES	125,00	125,00
<u>SOFTWARE DE CONTROL</u>		
Programa diseñado para visualizar en planta y sobre su monitor de PC, las Plazas Libres- Ocupadas del Parking Creado bajo Windows con gran sencillez de manejo, el diseño de plantas en pantalla, facilita la localización de plaza, así como su vistosidad de representación. Visualiza un vehículo sobre la plaza ocupada, simulando su estado.		
	195,00	195,00

SISGNOSTICOS DE LOCALIZACION

Módulos luminosos de indicación, estado de las plazas detectado por sensores.
Facilita la localización de las plazas libres al usuario a su paso por las zonas de estacionamiento, compuesto de placa con pilotos led de alta resolución rojo-verde.

150 UDES	15,20	2.280,00
----------	-------	----------

PANTALLAS DE LED CON INDICADOR N° DE PLAZAS LIBRES MAS TEXTOS INFORMATIVOS

Medidas : 80x15x10 , Color de LED rojo.

2 UDES	725,25	1.450,50
--------	--------	----------

CUADRO DE PROTECCIONES

1 UDES	120,00	120,00
--------	--------	--------

ENTUBADO Y CABLEADO

La distribución de sensores por todas las plaza de estacionamiento obliga a crear una red de comunicación entre todos los elementos, siendo necesario y según características de aparcamiento su instalación

Tubo PVC 15 mm
Manguera de 2 x 0,5 Apantallado
Manguera de 2 x 1 Apantallado

1 UDES	2,95	2,95
--------	------	------

TOTAL MATERIAL

17.010,45

INSTALACION, MONTAJE Y PUESTA EN MARCHA

1 UD.	1.020,63
-------	----------

18.031,08

3.5.4. Presupuesto para implantar los sistemas de comunicación entre componentes.

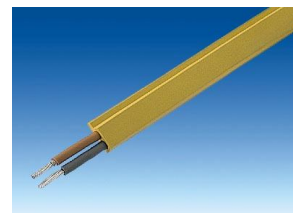
Como se explica en el apartado relativo a la selección del equipo, es necesario incorporar una serie de módulos para el funcionamiento del bus de campo seleccionado para llevar a cabo la comunicación entre los sensores/actuadores y el PLC. La empresa SIEMENS S.A. ofrece los

módulos necesarios para lograr la comunicación entre estos elementos mediante AS-Interface. La relación de precios es la que se muestra a continuación [25].

Componente	Precio unidad	Fotografía
FUENTE AS-INTERFACEL IP20; OUT: AS-I DC30V, 5A IN: AC 115V/230V C/DETECC. Defecto tierra integr. con detec. sobrecarga integr	321,08€	
SITOP Power 5 Modular, fuente de alimentación conmutada, entrada: 120-230-500 V AC, salida: 24V DC / 5A	157,30€	
SIMATIC NET, repetidor para interfase AS-i para la prolongación del cable, incluye pieza inferior FK-E	275,80€	
SIMATIC NET, extensor para interfase AS-i para el montaje distanciado del maestro (distancia hasta 100 m), incluye pieza inferior FK-E	306,60€	
SIRIUS NET, modulo compacto, AS-i, 4 salidas digitales (Y-II), M12, IP65/67	98,18€	

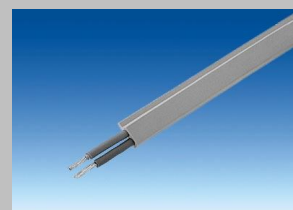
Conductor del AS-i, perfilado amarillo, cubierta de goma 2x	185,34€
--	----------------

1,5mm², 100m, 1 pieza = 100m
--



Conductor del AS-i, perfilado para tensión auxiliar externa	222,39€
--	----------------

hasta 24 V, negro, cubierta de goma 2 x 1,5mm², 100m, 1pieza = 100m



3.5.5 Presupuesto de la creación de la base de datos y del sistema de reserva vía web

Ya que el software necesario para la creación de ambas aplicaciones (base de datos y sistema de reserva vía web) es gratuito, sólo habría que considerar las horas dedicadas por el programador al Proyecto y el precio de la hora de trabajo. Considerando una precio de 30 €/hora y estimando el tiempo empleado para la creación de esta parte del Proyecto en unas 180 horas, el resultado del coste de la base de datos y el sistema de reserva vía web será de unos 5400 €.

De todos los presupuestos mostrados en los apartados anteriores cabe destacar que, salvo en los que se haya detallado explícitamente, ninguno de ellos incluye el I.V.A. ni la puesta en marcha.

Solución al Problema

4.1. Introducción

En este cuarto capítulo se va a elaborar una solución a la propuesta de automatización de un parking. En el ámbito de la automatización industrial pueden diferenciarse varios niveles de actuación reflejados en la pirámide CIM (Computer Integrated Manufacturing), tal y como se muestra en la Figura 4.1. A continuación se introduce una breve descripción de estos niveles.

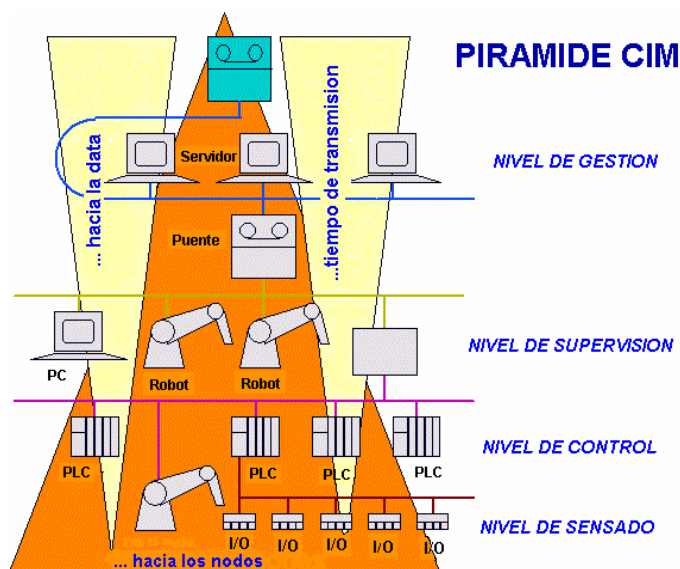


Figura 4.2. Pirámide CIM.

- ❖ Nivel de acción: Este nivel está formado por los elementos de medida o sensores, y los elementos de mando o actuadores, distribuidos en una línea de producción. Son los elementos más directamente relacionados con el proceso productivo, ya que los actuadores

son los encargados de ejecutar las órdenes de los elementos de control para modificar el proceso productivo (motores, válvulas, calentadores, etc.), y los sensores miden variables en el proceso de producción (nivel de líquidos, caudal, temperatura, presión, posición, etc.).

- ❖ Nivel de control: En este nivel se sitúan los elementos capaces de gestionar los actuadores y sensores del nivel anterior tales como autómatas programables o equipos de aplicación específica basados en microprocesador (robots, máquinas herramienta, o controladores). Estos dispositivos son programables y permiten que los actuadores y sensores funcionen de forma conjunta para ser capaces de realizar el proceso industrial indicado. Los dispositivos de este nivel juntos con los de nivel inferior de acción poseen entidad suficiente como para realizar procesos productivos por si mismos y es importante que posean buenas características de interconexión para ser enlazados con el nivel superior de supervisión, generalmente mediante el uso de buses de campo.
- ❖ Nivel de supervisión: En este nivel es posible visualizar como se están llevando a cabo los procesos de planta y a través de entornos SCADA poseer una “imagen virtual de la planta” de modo de que ésta se puede recorrer de manera detallada donde se muestren las posibles alarmas, fallos, alteraciones en cualquiera de los procesos que se llevan a cabo.
- ❖ Nivel de gestión: Este nivel se caracteriza por administrar la producción completa de la empresa, comunicar distintas plantas, mantener las relaciones con los proveedores y clientes, proporcionar las consignas básicas para el diseño y la producción de la empresa empleándose PCs, estaciones de trabajo y servidores de distinta índole.

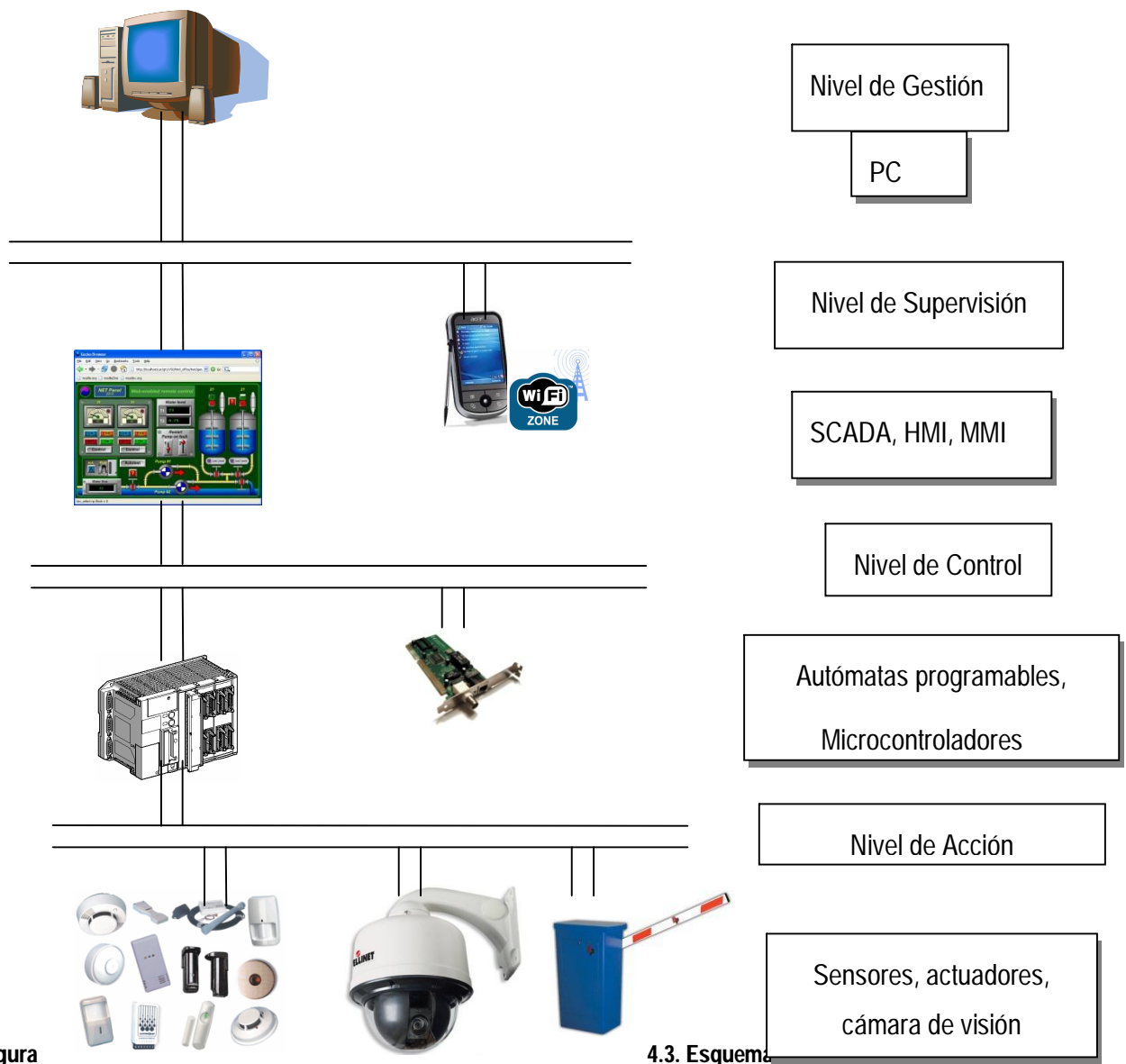
La estructura de pirámide CIM debe planificarse de “arriba hacia abajo”, es decir, dando prioridad a los niveles superiores, pero debe implementarse de “abajo hacia arriba”, es decir, empezando por el nivel más bajo [26].



Figura 4.2. Ejemplo pirámide CIM.

4.2. Implementación del sistema

En este apartado se describe el esquema de la organización y la estructura del sistema implementado (ver Figura 4.3).



Figura

4.3. Esquema

4.3. Especificación de Requisitos

4.3.1. Diagrama de transición de estados

El sistema está compuesto por los siguientes estados:

- ❖ Inicio Sistema → En este estado se realizan las operaciones necesarias para la puesta en marcha del sistema (alimentación, comprobación de sensores, etc.)

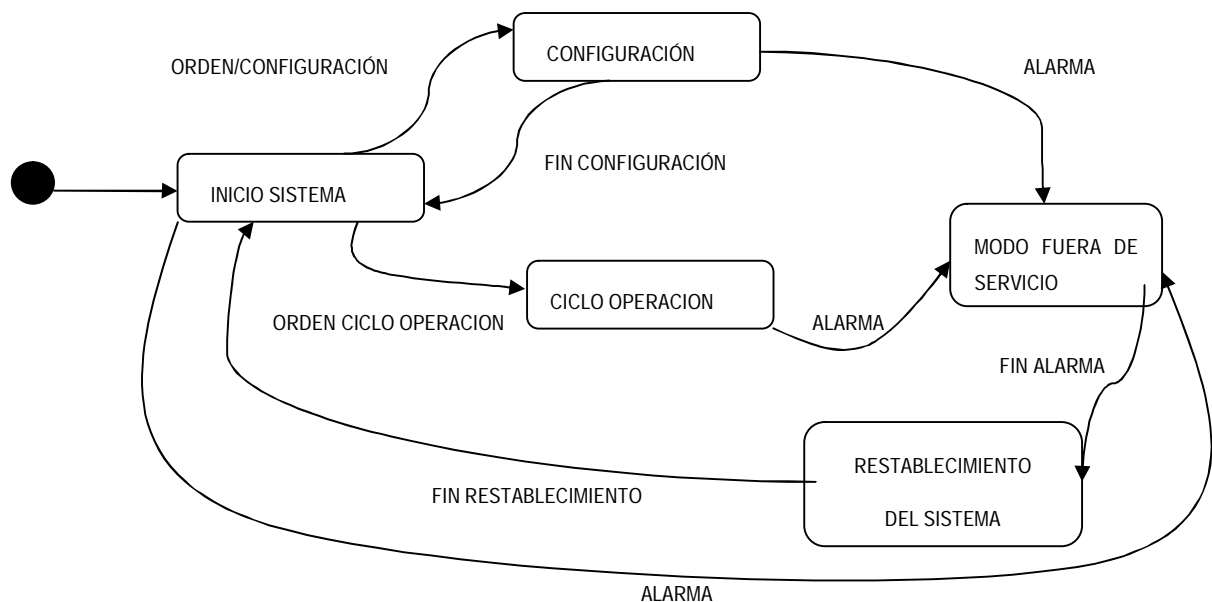
- ❖ Configuración → En este estado se configuran los parámetros del sistema.
- ❖ Ciclo de operación → En este estado se cargan las condiciones para el ciclo de operación, se implementa el funcionamiento del sistema.
- ❖ Modo fuera de servicio → Se produce cuando ocurre una incidencia, avería, limpieza de

Figura 4.4: Diagrama de transición de estados.

parking o tareas de mantenimiento, siendo imposible el acceso al parking.

- ❖ Restablecimiento del sistema → Una vez finalizado el modo fuera de servicio, este estado se encarga de restablecer el sistema dejándolo en el estado inicial.

4.3.2. Ciclo de operación y modo de funcionamiento



El ciclo de operación del sistema se divide en 4 estados: Inicio sistema, modo día, modo noche y modo fuera de servicio. El estado inicio de sistema, se encarga de la puesta en marcha inicial o arranque del sistema. El modo día corresponde a una banda horaria de funcionamiento concreta, con sus parámetros característicos, tales como funcionamiento de ambas plantas. En el modo noche, sólo se puede acceder a la planta pública. Cuando surgen averías o trabajos de mantenimiento, se carga el modo fuera de servicio, que evita el acceso al parking para la

correcta realización de trabajos y garantizar la seguridad de los clientes. En la figura 4.5 se pueden observar las relaciones entre los distintos estados.

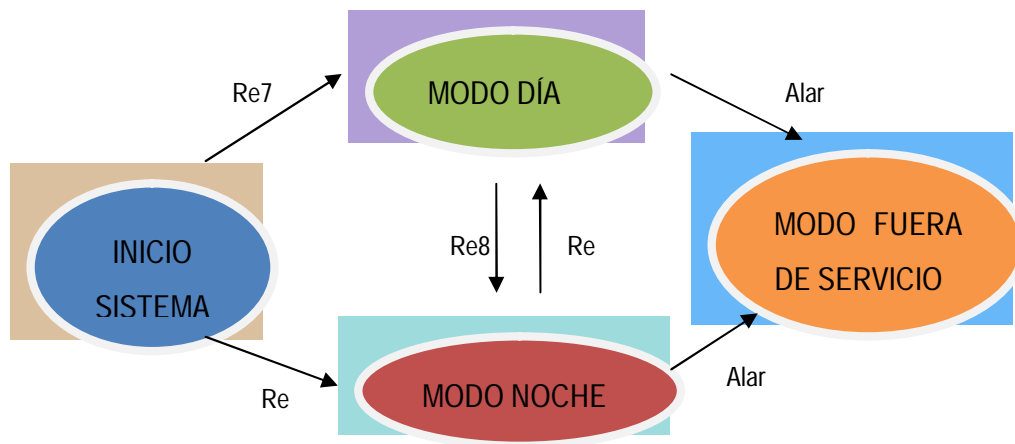
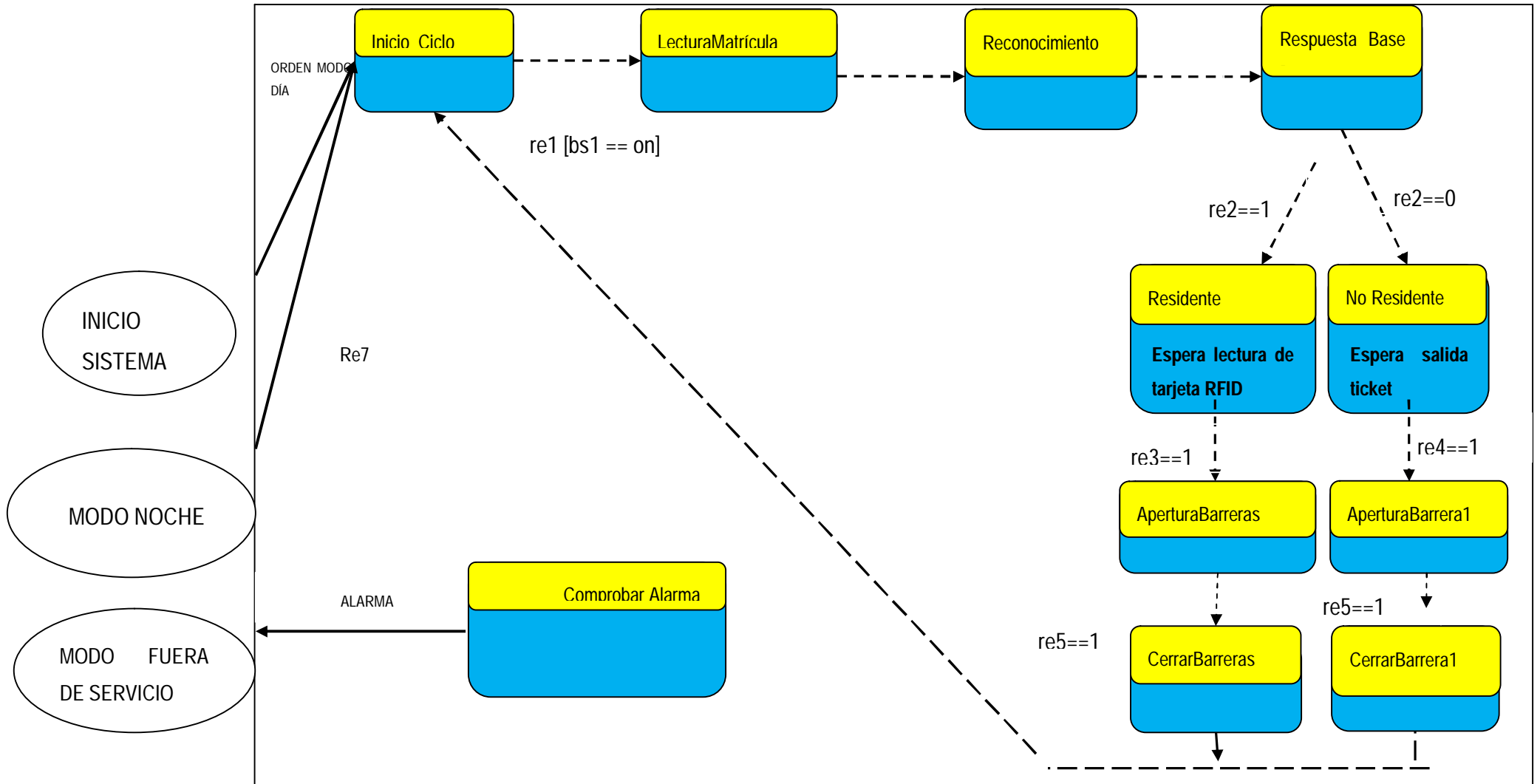
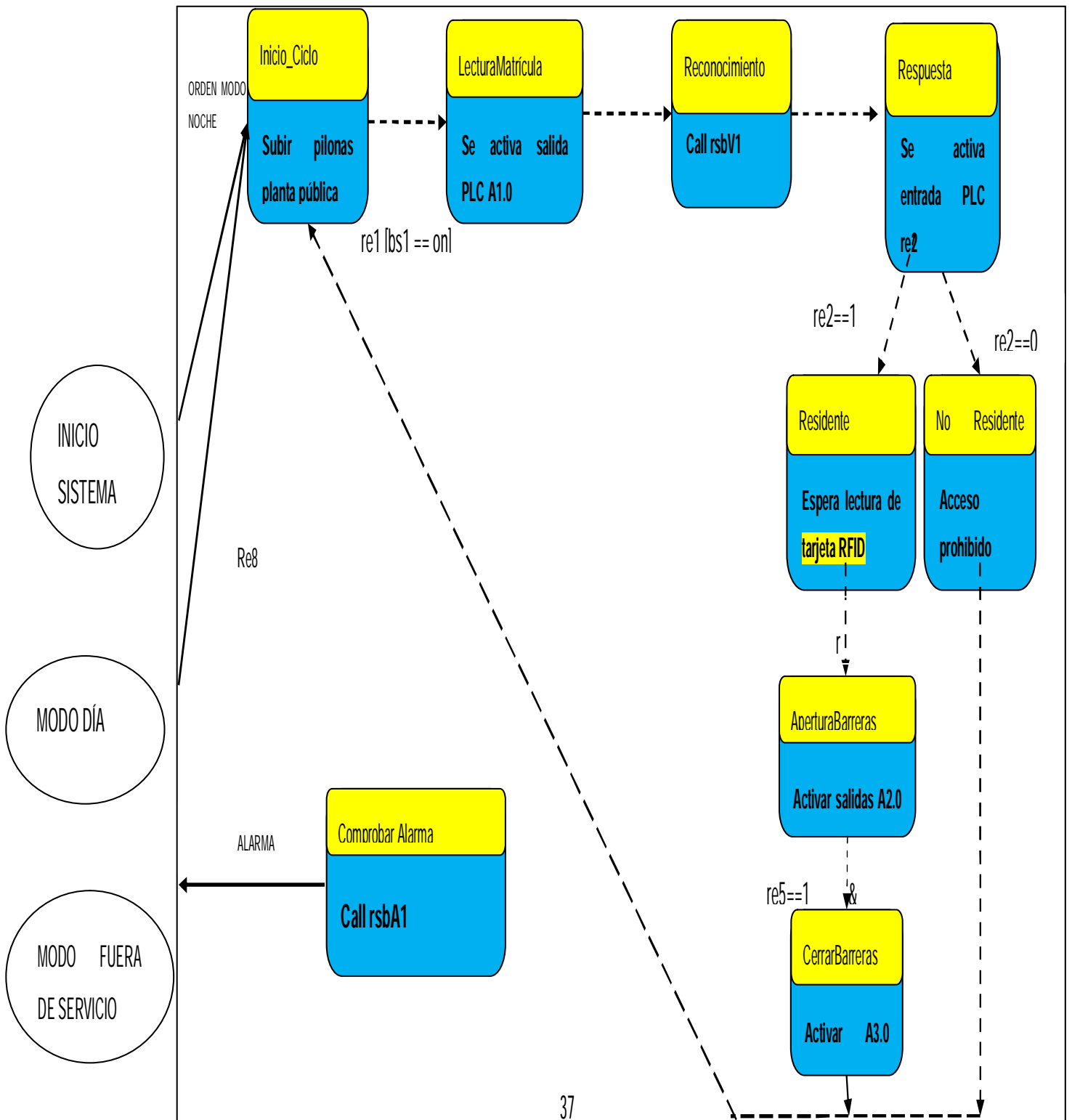


Figura 4.5. Ciclo de operación.

4. 3. 2. 1. Ciclo de operación modo día



4.3.2.2. Ciclo de operación modo noche



4.3.3. Tabla de descripción de estados

Estado	Acciones	Transiciones
Inicio Sistema	Descripción General: <ul style="list-style-type: none"> Habilitar el sistema y arrancarlo. 	<ul style="list-style-type: none"> Si Orden de Inicio sistema[Habilitado] → Modo día/Modo Noche Si Orden Configuración[Habilitado] → Configuración Si Alarma[Habilitado] → Modo fuera de servicio
Configuración	Descripción General: <ul style="list-style-type: none"> Configurar los parámetros del sistema Pre-condiciones: <ul style="list-style-type: none"> Sistema en estado inicio sistema 	<ul style="list-style-type: none"> Si Fin Configuración[] → Inicio Sistema Si Alarma [Habilitado] → Modo fuera de servicio
Modo día	Descripción General: <ul style="list-style-type: none"> Se realiza el proceso de funcionamiento principal para residentes y no residentes de 7:30 a 23:00 Pre-condiciones: <ul style="list-style-type: none"> Inicia el sistema 	<ul style="list-style-type: none"> Si Orden Modo Noche [Habilitado] → Modo Noche Si Alarma [Habilitado] → Modo fuera de servicio Si Fin modo fuera de servicio → Restablecimiento del Sistema
Modo noche	Descripción General: <ul style="list-style-type: none"> Se realiza el proceso de funcionamiento nocturno para residentes únicamente 	<ul style="list-style-type: none"> Si Fin modo fuera de servicio → Restablecimiento del Sistema Si Alarma[Habilitado] → Modo fuera de servicio Si Orden Modo Día [Habilitado] → Modo día
Restablecimiento del sistema	Descripción General: <ul style="list-style-type: none"> Se ha producido una parada de de emergencia. El sistema debe ser restablecido para que pueda funcionar correctamente. La alarma viene definida por una incidencia, servicio de mantenimiento, servicio de limpieza o reparación de averías. 	<ul style="list-style-type: none"> Si Fin Restablecimiento del Sistema → Inicio sistema
Modo fuera de Servicio	Descripción General: <ul style="list-style-type: none"> Funcionamiento parado por algún tipo de incidencia (Alarma). 	<ul style="list-style-type: none"> Si Fin modo fuera de servicio → Restablecimiento del Sistema.

Tabla 4.1. Descripción de los estados de primer nivel.

4.3.4. Tabla de subrutinas

Subrutina	Descripción
RsbV1	<ul style="list-style-type: none"> Determinación de la matrícula mediante el sistema OCR para reconocimiento de caracteres. Envío de dicha información a la base de datos para la comprobación Residente/no Residente
RsbA1	<ul style="list-style-type: none"> Cambio de estado a modo fuera de servicio mediante esta subrutina, la cual es activada cuando se produce una incidencia o alarma, ya sea, de mantenimiento, de reparación o de limpieza para impedir el acceso al parking.

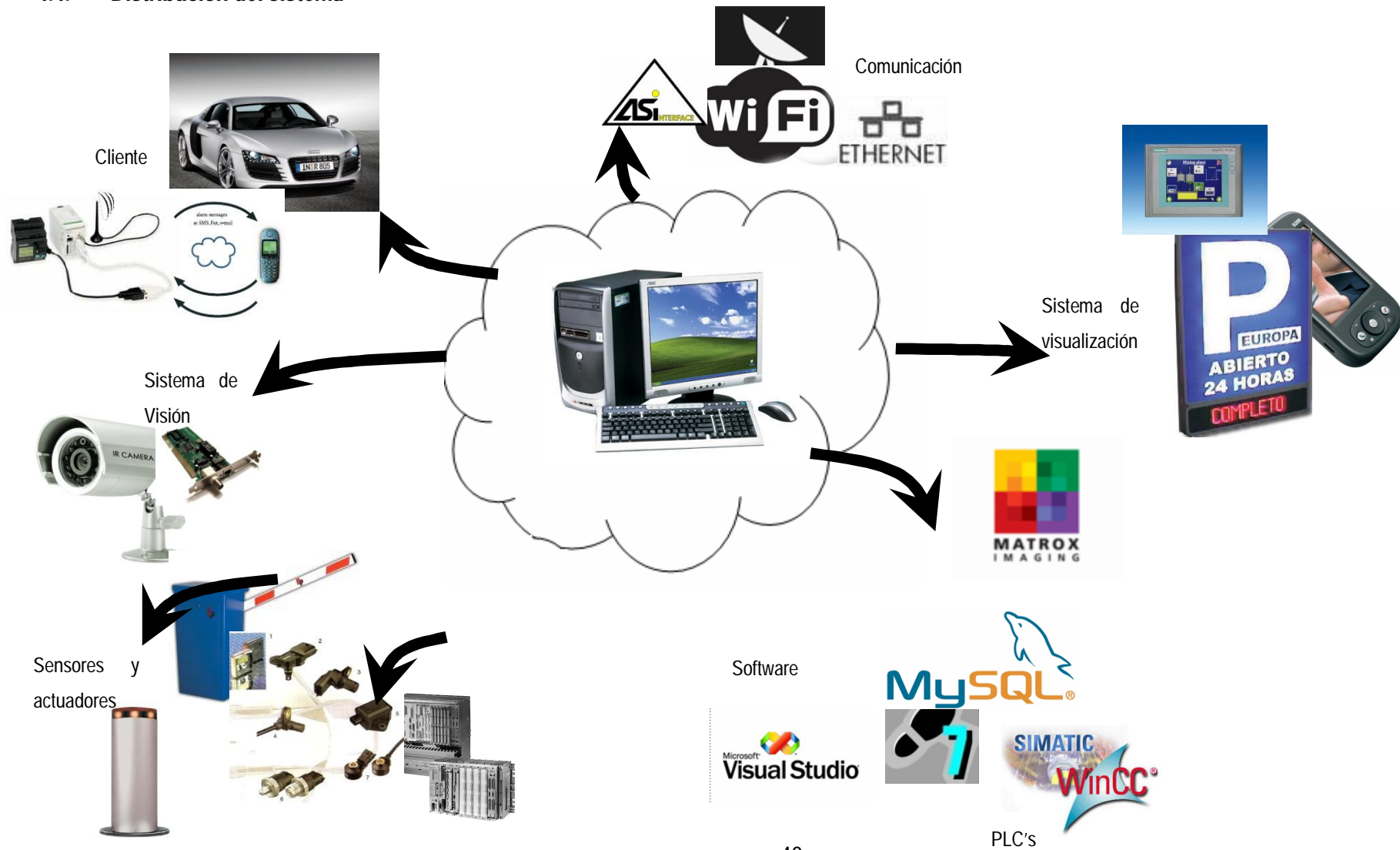
Tabla 4.2. Subrutinas

4.3.5. Tabla de descripción de eventos

Evento	Origen	Descripción/Significado
Re1	<ul style="list-style-type: none"> Cámara de visión. PLC 	<ul style="list-style-type: none"> Coche esperando para acceder al parking Preparado para lectura de matrícula
Re2	<ul style="list-style-type: none"> Base de datos. PLC 	<ul style="list-style-type: none"> Si su valor es 1→ Coche identificado por la base de datos como residente y preparado para lectura de RFID. Si su valor es 0→coche identificado por la base de datos como no residente y preparado para la entrega de ticket
Re3	<ul style="list-style-type: none"> PLC 	<ul style="list-style-type: none"> Tarjeta RFID leída por el sistema de identificación. Preparado para la apertura de las 2 barreras.
Re4	<ul style="list-style-type: none"> PLC 	<ul style="list-style-type: none"> Ticket de acceso entregado al cliente. Preparado para la apertura de barrera1
Re5	<ul style="list-style-type: none"> PLC 	<ul style="list-style-type: none"> Coche ya ha superado la barrera principal de entrada al parking. Preparado para la bajada de la barrera1
Re6	<ul style="list-style-type: none"> PLC 	<ul style="list-style-type: none"> Coche residente ya ha superado la barrera de acceso a la planta privada. Preparado para la bajada de la barrera de residentes.

Tabla 4.3. Lista de eventos.

4.4. Distribución del sistema



4.5. Descripción del software utilizado

4.5.1. Software de programación de autómatas

Simatic WinCC

Simatic WinCC (ver Figura 4.7) es el software HMI para todas las aplicaciones a nivel de máquina y de proceso en la construcción de instalaciones y maquinaria y en la construcción de máquinas en serie. El software de ingeniería permite la configuración continua y homogénea de todos los equipos SIMATIC HMI. *Simatic WinCC* abarca desde los Micro Panels configurados para aplicaciones con los autómatas SIMATIC S7 200 hasta soluciones “in situ” con SIMATIC Panels PCs y salas de control locales equipadas con PCs estándar.

WinCC flexible tiene máxima eficiencia de configuración de librerías con objetos preconfigurados, módulos de imagen reutilizables, herramientas inteligentes e incluso la traducción automatizada de textos en proyectos multilingüe.

WinCC flexible Runtime ofrece la funcionalidad básica HMI en PCs incluyendo un sistema de alarmas e informes y, de ser necesario, puede ser puntualmente ampliado.

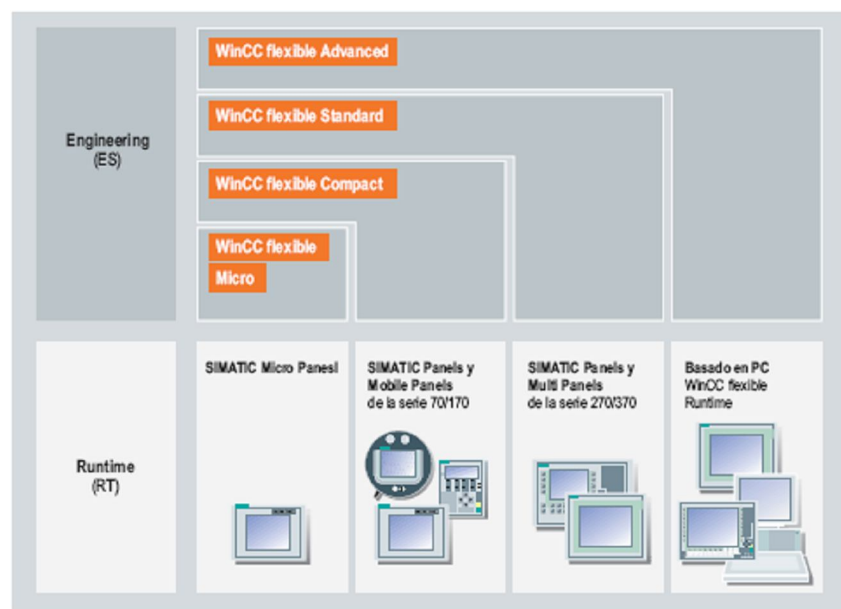


Figura 4.7. Simatic WinCC.

Las funciones runtime disponibles en los equipos SIMATIC HMI (Panels, Mobile Panels y Multi Panels) dependen del tipo de equipo que se adquiera [27].

WinCC flexible está disponible en diferentes variantes, escalonadas por precio y prestaciones, que se complementan y están perfectamente adaptadas a los distintos tipos de paneles. El paquete software siempre comprende la posibilidad de configuración del paquete más pequeño.

Simatic Step7 V5.2

STEP 7 V5.2 es el software estándar para configurar y programar los sistemas de automatización SIMATIC. STEP 7 forma parte del software industrial SIMATIC. El software estándar STEP 7 presenta las siguientes variantes:

- ❖ STEP 7-Micro/DOS y STEP 7-Micro/WIN para aplicaciones stand-alone sencillas en sistemas de automatización SIMATIC S7-200.
- ❖ STEP 7 para aplicaciones en sistemas de automatización SIMATIC S7-300/400, SIMATIC M7-300/400 y SIMATIC C7 con funciones ampliadas: Ampliable con los productos de software opcionales integrados en el Software Industrial SIMATIC. Posibilidad de parametrizar bloques de función y de comunicación, forzado y modo multiprocesador, Comunicación de datos globales, Transferencia de datos controlada por eventos con bloques de comunicación y de función y Configuración de enlaces.

En la Figura 4.8 se muestran las tareas básicas que se deben realizar en la mayoría de los proyectos.

4.5.2. Software de programación de la base de datos y comunicaciones con dispositivos móviles.

Microsoft Visual Studio

Microsoft Visual Studio (ver Figura 4.9) es un entorno de desarrollo integrado (IDE, por sus siglas en inglés) para sistemas Windows. Soporta varios lenguajes de programación tales como Visual C++, Visual C#, Visual J#, ASP.NET y Visual Basic .NET, aunque actualmente se han desarrollado las extensiones necesarias para muchos otros.

Visual Studio permite a los desarrolladores crear aplicaciones y sitios web, así como servicios web en cualquier entorno que soporte la plataforma .NET (a partir de la versión 6). Así, se pueden crear aplicaciones que se comuniquen desde distintas estaciones de trabajo, páginas web o dispositivos móviles.

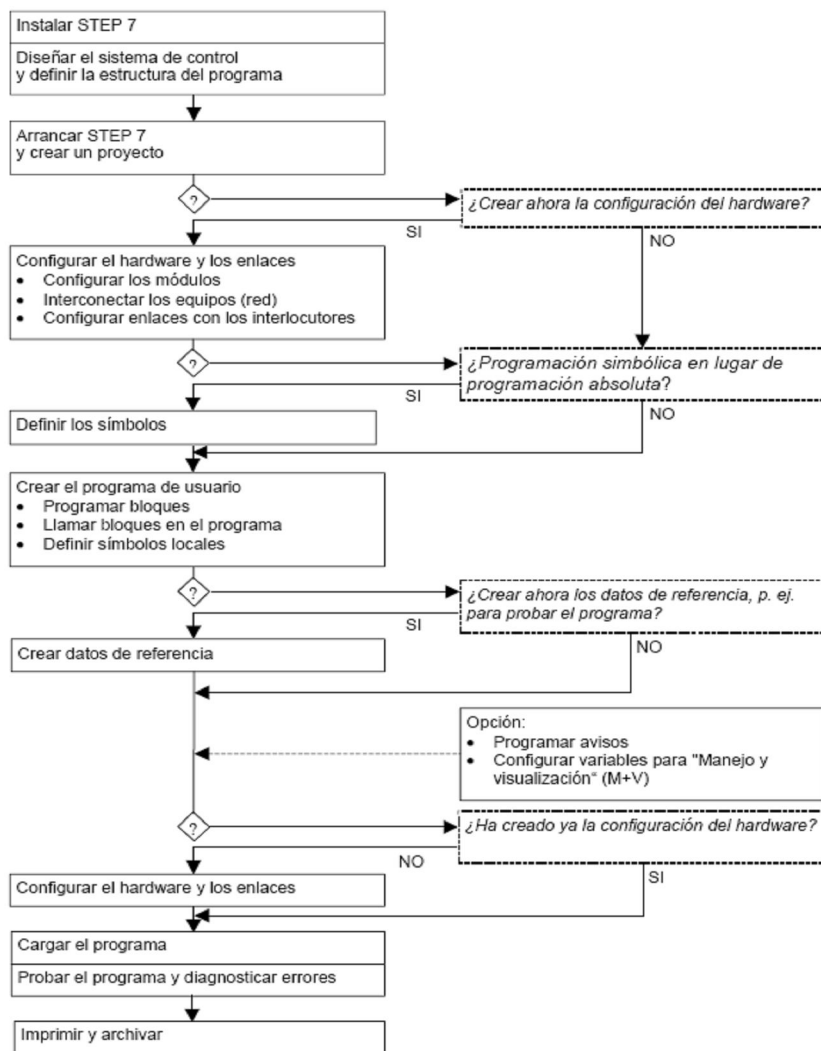


Figura 4.8. Tareas básicas en STEP 7.

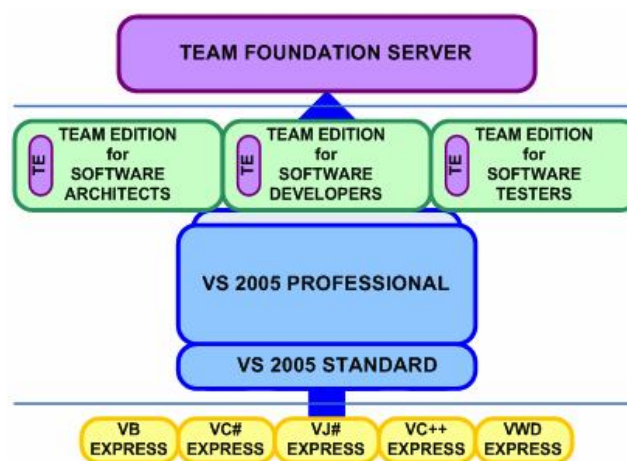


Figura 4.9. Esquema de Microsoft Visual Studio.

Visual Studio 2005 incluye un diseñador de implantación, que permite que el diseño de la aplicación sea validado antes de su implantación. También se incluye un entorno para publicación web y pruebas de carga para comprobar el rendimiento de los programas bajo varias condiciones de trabajo. También añade soporte de 64-bit. Aunque el entorno de desarrollo sigue siendo una aplicación de 32 bits, Visual C++ 2005 soporta compilación para x86-64 (AMD64 e Intel 64) e IA-64 (Itanium). El SDK incluye compiladores de 64 bits así como versiones de 64 bits de las librerías. Además, tiene varias ediciones radicalmente distintas entre sí: Express, Standard, Professional, Tools for Office, y cinco ediciones Visual Studio Team System. Éstas últimas se proporcionaban conjuntamente con suscripciones a MSDN, cubriendo los cuatro roles principales de la programación: arquitectos, desarrolladores de software, testers y profesionales de las bases de datos. La funcionalidad combinada de las cuatro ediciones Team System se ofrecía como la edición Team Suite [28].

SQL Server Management Studio

Management Studio (ver Figura 4.10) es una herramienta nueva para los administradores de bases de datos y programadores de SQL Server. Integrado en Microsoft Visual Studio, ofrece herramientas gráficas para la administración de bases de datos, junto con un completo entorno de desarrollo. Management Studio incluye en una sola herramienta las funciones Administrador corporativo, Analysis Manager y Analizador de consultas SQL de SQL Server 2000, junto con la posibilidad de escribir instrucciones MDX, XMLA y XML [29].



Figura 4.10. SQL Server.

4.5.3. Software de programación del sistema de visión

1) Librerías de Procesamiento de Imágenes MIL (Matrox Imaging Library)

Las librerías MIL son un paquete software, independiente del hardware, diseñadas para facilitar la labor de programación en el campo del procesamiento de imágenes. El paquete MIL está desarrollado en lenguaje C y disponible en forma de librería de funciones para Windows 95/98/NT, utilizando entornos de desarrollo como Microsoft Visual C, BorlandC++ o C++Builder. Entre las facilidades que nos ofrecen, se pueden resaltar las siguientes:

- ❖ Adquisición de imágenes desde un fichero o desde la cámara.
- ❖ Capacidad de procesamiento como el suavizado, filtrados espaciales y estadísticos
- ❖ Operaciones gráficas como la definición de una ventana, escritura de un texto, rectángulos, arcos, etc.
- ❖ Análisis de contornos / blobs.
- ❖ Capacidad para el reconocimiento de patrones.
- ❖ Reconocimiento de caracteres (OCR)
- ❖ etc.

El paquete MIL, tiene una profundidad o resolución que depende de la aplicación:

- ❖ Permite grabar imágenes de 8-bits en escala de grises o de 16-bits en color.
- ❖ Permite procesar imágenes de 8, 16 y 32 bits de profundidad.
- ❖ Permite procesar imágenes en color dependiendo de la operación.
- ❖ Las funciones de análisis estadístico, las operaciones de reconocimiento de caracteres (OCR) y reconocimiento de patrones, solo trabajan en escala de gris.

Las MIL ofrecen la posibilidad de crear aplicaciones de tipo consola, donde se abre una ventana MS-DOS con una ventana propia de las MIL. Sin embargo, también es posible realizar aplicaciones con interfaces gráficos, utilizando los controles más avanzados de los entornos Visual C++ o C++ Builder [30].

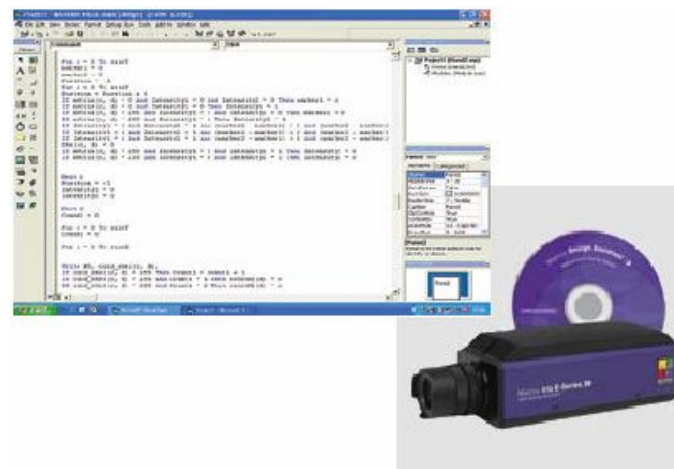


Figura 4.10. Matrox Imaging Library.

CAPÍTULO 5

SISTEMA DE RECONOCIMIENTO DE MATRÍCULAS MEDIANTE VISIÓN ARTIFICIAL

El reconocimiento automático de matrículas (Automatic Number Plate Recognition o ANPR) es un método de vigilancia que utiliza el reconocimiento óptico de caracteres (Optical Character Recognition u OCR) para leer las matrículas de los vehículos. Los ANPR se pueden utilizar para almacenar las imágenes capturadas por las cámaras, así como el texto de la matrícula. En algunos casos, es posible configurarlos para almacenar también una fotografía del conductor. Estos sistemas utilizan a menudo iluminación infrarroja para que la cámara pueda tomar fotografías en cualquier momento del día. La tecnología ANPR tiende a ser específica para un país o región, debido a la variación de matrículas entre unos lugares y otros[34].

5. 1. La Visión Artificial desde sus comienzos

La Visión Artificial puede definirse como “el proceso de extracción de información del mundo físico a partir de imágenes, utilizando para ello un ordenador” o como “la ciencia que estudia la interpretación de imágenes mediante computadores digitales”.

El propósito de la Visión Artificial es programar un ordenador para que “vea” y “entienda” una escena o las características de una imagen. En la Figura 5.1, quedan representadas las relaciones entre la Visión Artificial y otras áreas afines.

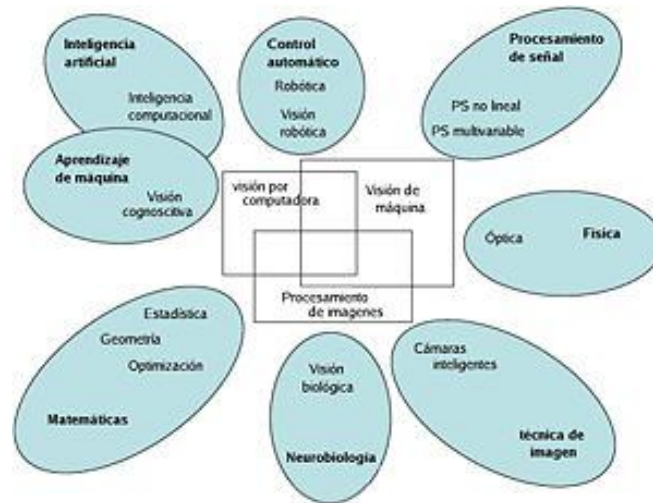


Figura 5.1. Relación entre la Visión Artificial y otras áreas afines.

Los objetivos típicos de la Visión Artificial incluyen:

- La detección, segmentación, localización y reconocimiento de ciertos objetos en imágenes.
- La evaluación de los resultados.
- Registro de diferentes imágenes de una misma escena u objeto, i.e., hacer concordar un mismo objeto en diversas imágenes.
- Seguimiento de un objeto en una secuencia de imágenes.
- Mapeo de una escena para generar un modelo tridimensional de la escena; tal modelo podría ser usado por un robot para navegar por la escena.
- Búsqueda de imágenes digitales por su contenido.

Estos objetivos se consiguen por medio de técnicas de reconocimiento de patrones, aprendizaje estadístico, geometría de proyección, procesado de imágenes, teoría de gráficos, etc. La Visión Artificial cognitiva está muy relacionada con la psicología cognitiva y la computación biológica.

La tecnología desarrollada entorno a la Visión Artificial desde finales de 1970, en la actualidad permite contar con múltiples herramientas para la mejora de la calidad de las imágenes. Algunas industrias no desarrollan sus productos haciendo uso de técnicas de Visión Artificial como parte íntegra de sus estrategias de fabricación. Sin embargo, para otros, la Visión Artificial representa un medio rentable y razonable que incrementa de modo significativo la calidad del producto final.

Sistema de reconocimiento automático de matrículas (ANPR)

Este apartado describe la implementación del sistema de reconocimiento de matrículas mediante OCR (ver figura 5.2). La implementación de este sistema es de obligado cumplimiento en España puesto que, según la Ley 14/11 2002 B.O.E., es obligatoria la impresión en el ticket de entrada de la matrícula y hora de llegada de cada vehículo al parking.

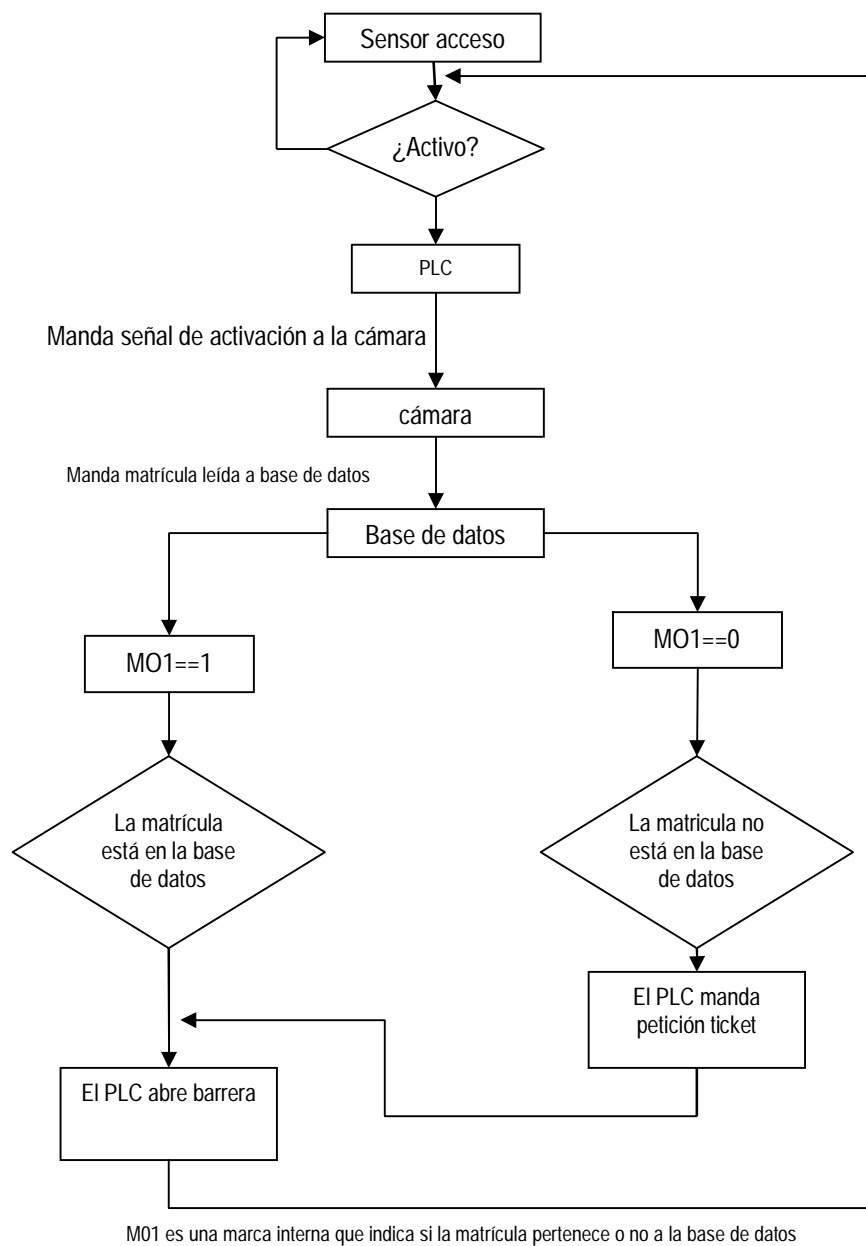


Figura 5.2

El sistema de reconocimiento de matrículas aporta numerosas ventajas entre las que cabe destacar: menores tiempos de espera para los usuarios registrados, mejor control de

entrada/salida de vehículos, mayor seguridad en el parking, interacción con otros organismos tales como Policía, Ayuntamientos o Dirección General de Tráfico, optimización en la gestión de sanciones por parte de la empresa o reclamaciones por parte del cliente, etc. Por todo esto, la implantación de sistemas ANPR en España va en aumento desde hace unos años.

Los sistemas ANPR suelen estar compuestos por los siguientes elementos: Unidad de Control, Unidad de Adquisición y Unidad de Procesado, todos ellos descritos brevemente a continuación.

Unidad de Control

La unidad de control se encarga de regular el proceso de acceso al parking: desde el control de plazas libres, hasta el accionamiento de la barrera, entre otras acciones. La unidad de control obtiene información de los distintos sensores que tiene el sistema como, por ejemplo, los sensores de entrada/salida, los sensores de detección de humos, los sensores de ocupación de plazas, los sensores de llegada de un vehículo, etc.

La unidad de control está compuesta por:

- Un bus de comunicaciones que integra los diferentes elementos del sistema.
- Un PLC Siemens modelo S7 serie 300.
- Un módulo de ampliación de entradas/salidas para el PLC, para la conexión de los diferentes sensores y actuadores del sistema.
- Varios sensores inductivos para detectar el paso de los vehículos.
- Varios sensores ultrasónicos para detectar cuándo están ocupadas las distintas plazas.
- Varios sensores de espiras magnéticas para detectar de llegada de un vehículo.
- Varios actuadores (motores, barreras automáticas, pilonas de acceso, etc.).

Unidad de Inspección

La unidad de inspección tiene la tarea de suministrar al operador información visual del vehículo, una vez éste se encuentra situado a la entrada del parking. El sistema realiza una grabación continua del acceso al parking hasta el momento en el que llega un vehículo. Entonces, se realiza una captura fija de la parte baja de dicho vehículo para obtener información de su matrícula. La unidad de inspección está compuesta de:

- Cámara de color digital AXIS 211(ver Figura 5.3a) con interfaz IEEE802.3af (Ethernet). Esta cámara tiene una resolución de 640x480 y 25 fps (frames per second). Para el montaje del sistema se utiliza el modelo Basler A601fc (ver Figura 5.3b) con interfaz de comunicación IEEE1394 (Firewire) y una resolución de 640x480 y 60 fps.
- Sistema de iluminación. La cámara seleccionada tiene un sistema de infrarrojos integrado capaz de obtener la captura en condiciones de 0 lux (oscuridad total), por lo que el sistema carece de iluminación adicional para la captura. Sin embargo, sí existe una fuente de iluminación externa al parking situada en la rampa de acceso para facilitar la maniobra de los vehículos. Para el montaje de la maqueta se ha utilizado luz fluorescente con un variador de frecuencia para evitar la captura de barrido de frecuencia del tubo, con el fin de obtener una imagen más nítida.
- Sistema de comunicación. Se utiliza una conexión Ethernet con el PLC. En el montaje del sistema se emplea una conexión Firewire (estándar IEEE1394) con el PC.
- Frame grabber Matrox Meteor II, para adquisición de datos.

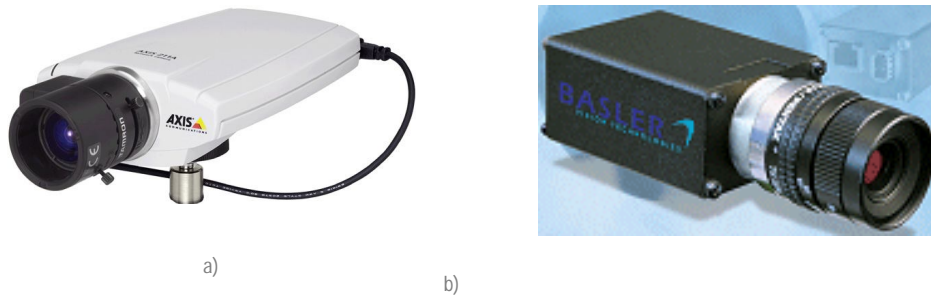


Figura 5.3. (a) Cámara de color digital AXIS 211, (b) Cámara Basler A601fc.

5.1.1.1. La tarjeta Matrox Meteor II

Para que la cámara pueda interactuar con el ordenador (éste será el encargado de procesar las imágenes obtenidas por la cámara), se necesita de una tarjeta controladora de video, que realice parte del procesamiento de imágenes.

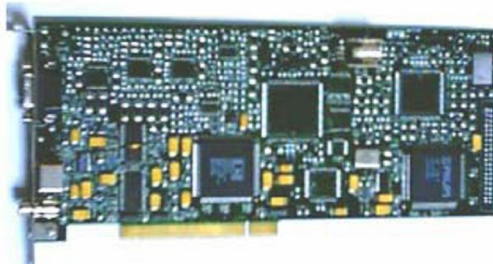


Figura 5.4

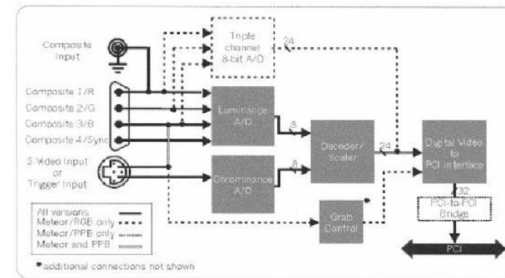


Figura 5.5

Al mismo tiempo, esta tarjeta proporciona al sistema una entrada de video para conectar la cámara. Inicialmente se usó una tarjeta *Matrox Meteor*. La tarjeta *Matrox Meteor*, es una tarjeta que permite operar tanto con imágenes en blanco y negro (B/N) como en color. Con esta tarjeta se puede mostrar la imagen en tiempo real y se pueden almacenar secuencias en memoria, en función de la velocidad del sistema en que se instala. Entre las muchas posibilidades que ofrece la tarjeta, cabe destacar la posibilidad de mostrar en una ventana, vídeo directo obtenido desde la cámara. Esta utilidad servirá para poder visualizar en el ordenador lo que está captando la cámara en tiempo real. A la hora de capturar vídeo, la tarjeta admite tanto *RS-170/CCIR* monocromo como *NTSC/PAL* en color y dispone de la posibilidad de conectar hasta cuatro dispositivos de captura de imágenes a la misma tarjeta simultáneamente. Para mayor facilidad en la calibración y la configuración de la tarjeta, es posible también pre-programar el brillo, el contraste y la saturación de los colores de las ventanas de captura mostradas[33].

La tarjeta *Matrox Meteor* es una tarjeta capturadora o *frame grabber* que puede utilizarse tanto con cámaras monocromo como color y que puede transferir tanto imágenes como vídeo en tiempo real.

Las características de la tarjeta son:

- Puede capturar imágenes en color, en formatos tan diferentes como NTSC, PAL, SECAM, RS170, o CCIR.
- Se conecta a puertos PCI.
- Puede transferir datos en tiempo real al sistema o almacenarlos en zonas de memoria.
- Tiene múltiples entradas de vídeo (hasta cuatro canales).

- Unidades de escala de vídeo de alta calidad.
- Sincronización estable.
- Puede trabajar con Matrox MGA Millenium4I o cualquier VGA.
- Se distribuye con un software compuesto por las librerías MIL (Martos Imaging Library), MIL-Lite, y Matrox Inspector.
- Soporta Windows NT, Windows 95, y DOS4GW 32-bit DOS

Los interfaces y conectores que soporta son:

- Interfaz PCI 32-bit.
- Conector de vídeo (DB9 para entradas de vídeo o señales RGB).
- Conector separado de entrada Y/C (4 pin mini din).
- Entrada para Phono Jack.

El consumo de la tarjeta es:

- Potencia consumida < 7.5W.
- Tamaño de la tarjeta 23.55cm x 10.55cm (9.3" x 4.2")
- Temperatura de trabajo desde 0°C a 55°C (32°F - 131°F)
- Humedad relativa hasta el 95% (no condensación)

5.1.1.2. La cámara digital AXIS 211

La cámara AXIS 211 es una cámara de red de nivel profesional para aplicaciones de vigilancia y supervisión remota en interiores y exteriores. Ofrece una de las funcionalidades más extensas del sector y es una elección perfecta para velar por la seguridad de oficinas, comercios y otros establecimientos a través de una red de área local o de Internet.

La cámara AXIS 211 está diseñada para proporcionar la mejor calidad de vídeo de su categoría y utiliza un sensor CCD de exploración progresiva y un eficaz hardware de procesamiento de imágenes en tiempo real para garantizar la máxima frecuencia de imagen incluso con resolución VGA. Además, su exclusiva combinación simultánea de flujos de compresión de vídeo Motion JPEG y MPEG-4 avanzado, permite optimizar los sistemas para obtener la mejor calidad de imagen y el uso más eficiente del ancho de banda.

La conexión directa a redes IP a través de Ethernet, junto con el servidor Web incorporado, la interfaz de aplicaciones abiertas y sus funciones avanzadas de acceso a la red basadas en estándares de la industria, simplifican enormemente la instalación y la integración de estos sistemas. Esta cámara de red es perfecta para la visualización, grabación y administración de vídeo.

A continuación, en la tabla 5.1, se describen las características técnicas de esta cámara.

Modelo	AXIS 211: Alimentación a través de Ethernet, objetivo varifocal con iris de tipo DC, interiores/exteriores
Sensor de imagen	Sensor CCD de barrido progresivo de 1/4" RGB Sony Wfine
Objetivo	AXIS 211: 3.0 - 8.0 mm, F1.0, iris de tipo DC, montura CS
Ángulo de visión	AXIS 211: horizontal 27°-67°
Iluminación mínima	AXIS 211: 0,75 lux, F1.0
Compresión de vídeo	Motion JPEG MPEG-4 Parte 2 (ISO/IEC 14496-2), Perfiles: ASP y SP
Resoluciones	16 resoluciones desde 640 x 480 hasta 160 x 120 vía API, 5 selecciones vía página Web de configuración
Frecuencia de imagen	Motion JPEG: Hasta 30 imágenes por segundo en todas las resoluciones MPEG-4: Hasta 25 imágenes por segundo a 640x480 Hasta 30 imágenes por segundo a 480x360 más bajas
Transmisión de vídeo	Motion JPEG y MPEG-4 simultáneos Frecuencia de imagen y ancho de banda controlables Frecuencia de bit variable y constante (MPEG-4)
Ajustes de la imagen	Niveles de compresión: 11 (Motion JPEG)/23 (MPEG-4) Rotación: 90°, 180°, 270° Nivel de color configurable, brillo, contraste, exposición, balance de blancos, configuración más precisa del comportamiento con poca luz Funciones de texto en pantalla: hora, fecha, máscara de privacidad, texto o imagen
Velocidad de obturación	De 2 segundos a 1/12500 segundos
Seguridad	Protección multiusuario mediante contraseña para restringir los niveles de acceso a la cámara y filtro de dirección IP
Usuarios	Hasta 20 usuarios simultáneos Número ilimitado de usuarios en multidifusión (MPEG-4)
Soporte de lenguajes (Interfase Web)	Ingles. Archivos de lenguajes descargables para Francés, Alemán, Italiano, Japonés y Español están disponibles en www.axis.com/techsup Otros archivos de lenguajes pueden estar disponibles
Gestión de eventos y alarmas	Eventos activados por detección de movimiento integrada, entrada externa o según una planificación Carga de imágenes a través de FTP, correo electrónico y HTTP Carga de imágenes a través de FTP, correo electrónico y HTTP Notificación a través de TCP, correo electrónico, HTTP y salida externa

Memoria previa y posterior a la alarma 1,2 MB (hasta 40 segundos de vídeo con resolución de 320x240 a 4 imágenes por segundo)

Conectores	Ethernet 10BaseT/100 BaseTX, RJ-45 Bloque de terminal para E/S (1 entrada de alarma, 1 salida) y conexión de alimentación eléctrica alternativa
Procesadores, memoria y reloj	CPU: ETRAX 100LX 32bit Procesamiento y compresión de vídeo: ARTPEC-2 RAM: 16 MB, Flash: 4 MB Reloj de tiempo real con batería
Alimentación	7-20 V DC 7 W máx. AXIS 211: alimentación a través de Ethernet (IEEE 802.3af) con clasificación eléctrica: Clase 2
Condiciones de funcionamiento	5 - 45 °C (41 - 113 °F) Humedad relativa: 20 - 80 % (sin condensación)
Instalación, gestión y mantenimiento	Herramienta AXIS Camera Management en CD y configuración basada en la Web Configuración de backup y restablecimiento Actualizaciones de firmware a través de HTTP o FTP, firmware disponible en www.axis.com
Acceso a vídeo desde el navegador Web	Visualización en directo Grabación de vídeo en archivo (ASF) Recorrido de secuencias para un máximo de 20 fuentes de vídeo Axis Páginas HTML personalizables
Requisitos mínimos para uso a través de la Web	CPU Pentium III a 500 MHz o superior, o AMD equivalente 128 MB de RAM, tarjeta gráfica AGP, 32 MB de RAM, Direct Draw, Windows XP, 2000, Server 2003, DirectX 9.0 ó posterior Internet Explorer 6.x o posterior Para otros sistemas operativos y navegadores consulte en www.axis.com/techsup
Integración de sistema	API abierta para la integración de la aplicación, con API AXIS VAPIX*, kit de desarrollo AXIS Media Control*, datos de activación de sucesos en transmisión de vídeo Calidad de Servicio (QoS) Capa 3, DiffServ Model Sistema operativo Linux incorporado * Disponible en www.axis.com
Protocolos compatibles	IPv4/v6, HTTP, TCP, QoS, RTSP, RTP, UDP, IGMP, RTCP, SMTP, FTP, ICMP DHCP, UPnP, Bonjour, ARP, DNS, DynDNS, SOCKS. Para más información acerca del uso de protocolos, consulte la página Web: www.axis.com
Software de gestión de vídeo (no incluido)	AXIS Camera Station: Aplicación de vigilancia para visualización, grabación y archivo de hasta 25 cámaras Aplicaciones de software disponibles a través de partners: www.axis.com/partner/adp_partners.htm
Accesorios (incluidos)	Fuente de alimentación de 9V DC, soporte, kit de conexión, guía de instalación, CD con herramienta de instalación, software y manual del

	usuario, licencias MPEG-4 (1 codificador, 1 decodificador), descodificador MPEG-4 (Windows)
Accesorios (no incluidos)	Carcasas para instalación en exteriores o en entornos adversos. Alimentación a través de midspans Ethernet Descodificador de vídeo en red AXIS 292 Pack con licencia multiusuario para decodificador MPEG-4
Homologaciones	EN 55022 Clase B, EN 61000-3-2, EN 61000-3-3, EN 55024, FCC Parte 15 Subparte B Clase B, ICES-003 Clase B, VCCI Clase B, C-tick AS/NZS 3548, EN 60950
Dimensiones (Alt. x Anch. x Prof.) y peso	Fuente de alimentación: EN 60950, UL, cUL AXIS 211: 38 x 95 x 178 mm (1.5" x 3.7" x 7.0") 250 g (0.55 lb) sin fuente de alimentación

Tabla 5.1. Características técnicas de la cámara digital AXIS 211.

Unidad de Procesado

La unidad de procesamiento de imágenes está formada por un PC con procesador Pentium IV a 2Ghz con 1 Gb RAM y un frame grabber de Matrox con salida IEEE1394. La gestión del frame-grabber para realizar la adquisición de imágenes se realiza mediante las librerías de procesado Matrox MIL (Matrox Imaging Library), programadas en lenguaje C/C++. Mediante estas librerías se realiza la grabación continua de la entrada del parking hasta que llega un vehículo. Cuando esto ocurre, se realiza la captura de una imagen fija, se calibra y, mediante un algoritmo de OCR, se obtiene la cadena de caracteres que aparecen en la matrícula.

5.1.1.3. Las librerías MIL

Las Matrox son unas tarjetas de adquisición de vídeo de prestaciones medias. Muchas compañías ofrecen tarjetas parecidas a las Matrox. Sin embargo, lo que las hace tan especiales, es el software que se vende con ellas, capaz de interactuar con la tarjeta aprovechando al máximo su rendimiento. Este software está compuesto por un conjunto de librerías denominadas MIL (ver Figura 5.6).

Estas librerías ofrecen un paquete de funciones de procesamiento de imágenes, totalmente independiente del hardware en el que se instalen y plenamente modular. Las MIL proporcionan una gran cantidad de funciones relacionadas con el tratamiento de imágenes, algunas de ellas especializadas para el análisis global, la obtención de patrones, o el reconocimiento de caracteres, entre muchas otras. Las MIL también proporcionan un conjunto básico de herramientas gráficas.

El programa desarrollado, se ha diseñado con la intención de que sea una herramienta sencilla, con un tiempo de aprendizaje corto, capaz de ser usado en cualquier plataforma y con procesos de gestión de imágenes completamente transparentes. Esto significa que las aplicaciones implementadas con MIL pueden funcionar en cualquier sistema que tenga instalada una tarjeta VGA compatible con el estándar VESA o, en su defecto, una tarjeta gráfica Matrox. Puesto que las MIL permiten la identificación unívoca de las tarjetas instaladas en el sistema, una misma aplicación puede ser capaz de controlar varias tarjetas al mismo tiempo. Los programas desarrollados con estas librerías, pueden interactuar con el microprocesador del sistema de manera segura y eficiente, beneficiándose de los sistemas de aceleración por hardware de las tarjetas Matrox. Todo ello redundará en una mayor eficiencia y una menor carga de trabajo para el micro, que podrá dedicarse a otras tareas, mientras la tarjeta gráfica se encarga del procesamiento de imágenes.

Las funciones incluidas en las librerías MIL se han programado pensando en un rendimiento optimizado del sistema, y utilizan las mejoras ya consideradas en la aceleración de gráficos de Intel MMX. Como ejemplo, baste decir que en un sistema Pentium II con Intel MMX, las operaciones de las MIL se pueden ejecutar hasta un 400% más rápido que en un microprocesador igual sin MMX [32]



Figura 5.6. Matrox Imaging Library (MIL)

Desarrollo de la aplicación de Visión Artificial

En este apartado se describe la implementación del sistema de Visión, haciendo hincapié en las funciones específicas del programa y en cómo se han resuelto los problemas surgidos durante la realización del mismo. La implementación del programa se ha desarrollado en C debido a las ventajas que suponía el uso de este lenguaje para incorporar la funcionalidad ofrecida por las

librerías MIL. Sin embargo, finalmente ha sido necesario re-implementar la aplicación en C++, ya que, de otro modo, resultaba imposible la integración del sistema de visión con el resto de aplicaciones desarrolladas para el control del parking.

La aplicación de reconocimiento de matrículas realiza los procesos que se enumeran a continuación y que se detallan en los apartados siguientes:

- Captura de la imagen digital de la matrícula.
- Adecuación de la imagen capturada para su análisis (pre-procesamiento).
- Procesamiento y análisis de la imagen.
- Obtención y posterior estudio de los resultados.

5.1.1.4. Captura de la imagen digital de la matrícula

Para la captura de la matrícula se ha optado por una cámara digital color con el fin de obtener una imagen tratable dentro de unos márgenes de calidad. Para ello, se montó un sistema de reconocimiento formado por:

- Una selección de balastos y tubos fluorescentes para obtener la intensidad lumínica necesaria para una correcta captura de la imagen.
- Un soporte regulable para la realización de distintas pruebas.
- Un montaje de cámara en soporte regulable para obtener la distancia programada de la parada del vehículo a la cámara.
- Una conexión Firewire entre la cámara y el frame-grabber .

La primera parte del código consiste en, una vez definidas e inicializadas las variables necesarias, reservar un espacio de memoria para la aplicación y el sistema, así como los distintos elementos del programa (displays, cámaras, buffers de imágenes, etc.). Todo esto se consigue utilizando la función `MappAllocDefault` ofrecida por las librerías MIL (ver extracto de código mostrado en la Figura 5.7).

Lo siguiente es obtener una imagen de la matrícula del coche que quiere acceder al parking. Utilizando las funciones `MdigGrabContinuous`, `MdigHalt` y `MdigGrab` se consigue, respectivamente: (1) la grabación continua de una secuencia de video, (2) la parada de dicha grabación y (3) la adquisición de una imagen fija de la zona de trabajo.

```

/* Localizaciones de memoria por defecto */

MappAllocDefault(M_SETUP, &MilApplication, &MilSystem,
                  &MilDisplay, &MilDigitizer, &MilImageDig);

/* Grabacion continua de entrada al parking */

MdigGrabContinuous(MilDigitizer, MilImageDig);

/* Al pulsar se congelará la imagen */

printf("\nAdquisicion de imagen preparanda:\n");
printf("-----\n\n");
printf("Grabacion continua en proceso.\n");
printf("Pulsa para parar\n\n");
getch();

/* Grabacion parada. */

MdigHalt(MilDigitizer);

```

Figura 5.7. Extracto de código relacionado con la inicialización de variables y el inicio de la grabación en continuo de la entrada del parking.

Una vez obtenida la imagen a tratar, se carga en memoria el patrón de calibración de caracteres utilizando la función `MbufRestore` y se visualiza en el display con `MdispSelect`, `Mdispcontrol` y `MdispIquire`. Ésta última función se encarga de obtener el tamaño adecuado del display (ver Figura 5.8).

```

/* Mostrando resultado */

printf("Grabacion continua parada\n\n");
printf("Pulsa para adquirir imagen\n\n");
getch();

/* La imagen adquirida se envia a buffer */

MdigGrab(MilDigitizer, MilImageDig);

/* Mostrando resultados */

printf("La imagen obtenida es:\n");
printf("Pulsa para finalizar\n\n");
getch();

/* Se carga la imagen patron*/

MbufRestore(IMAGE_FILE_DEFINITION, MilSystem, &MilImage);

/* Se dispone a visualizar la imagen en un único display */
MdispSelect(MilDisplay, MilImage);
MdispControl(MilDisplay, M_OVERLAY, M_ENABLE);
MdispInquire(MilDisplay, M_OVERLAY_ID, &MilOverlayImage);

```

Figura 5.8. Extracto de código encargado de mostrar la imagen capturada, así como el patrón de caracteres.

5.1.1.5. Pre-procesamiento de la imagen

La siguiente parte del código (ver Figura 5.9) se encarga de asignar localizaciones para las cadenas de caracteres (`MstrAlloc`, `MstrAllocResult` y `MstrControl`) y compara la imagen introducida como patrón con el texto de calibración definido anteriormente mediante la función `MstrEditFont`. Utilizando la función `MgraColor`, se permite la visualización de la imagen calibrada utilizando el patrón y sobre-escribiendo los caracteres reconocidos por la función `MstrEditFont`.

```
MstrAlloc( MilSystem, M_FEATURE_BASED, M_DEFAULT, &MilStrContext);
/* Se reserva memoria para la cadena de control resultado */
MstrAllocResult(MilSystem, M_DEFAULT, &MilStrResult);

MstrControl(MilStrContext, M_CONTEXT, M_FONT_ADD, M_USER_DEFINED);
/* Se comprueba la imagen patron con la cadena introducida */
MstrEditFont(MilStrContext, M_FONT_INDEX(0), M_CHAR_ADD,
M_USER_DEFINED + M_FOREGROUND_BLACK,
MilImage, TEXT_DEFINITION, M_NULL);

/* Se pintan los caracteres encontrados en la imagen patron */
MgraColor(M_DEFAULT, M_COLOR_GREEN);
MstrDraw(M_DEFAULT, MilStrContext, MilOverlayImage, M_DRAW_CHAR,
M_FONT_INDEX(0), M_NULL, M_ORIGINAL);
```

Figura 5.9. Extracto de código relacionado con la calibración de la imagen utilizando el patrón de caracteres y la sobre-impresión de la cadena identificada en la matrícula.

Con la función `MstrControl`, se introduce una cadena de caracteres y se indica el número de sub-cadenas que la componen (en este caso sólo una). Al mismo tiempo, con la función `MstrSetConstraint`, se definen los parámetros de la cadena a leer, entre los que destacan:

- Tipo de caracteres en cada posición de la cadena.
- Letras mayúsculas.
- Número mínimo de caracteres que deberán identificarse en la imagen de entrada.
- Número máximo de caracteres que podrán identificarse en la imagen de entrada.

```

MstrControl(MilStrContext, M_CONTEXT, M_STRING_NUMBER, 1);

/* Numero de caracteres max y min de la cadena a leer */
MstrControl(MilStrContext, M_STRING_INDEX(M_ALL), M_STRING_SIZE_MIN, 7);
MstrControl(MilStrContext, M_STRING_INDEX(M_ALL), M_STRING_SIZE_MAX, 8);

/* Solo letras mayusculas */
MstrSetConstraint(MilStrContext, M_STRING_INDEX(0), M_DEFAULT, M_LETTER + M_UPPERCASE, M_NULL);

/* Se fijan el tipo de caracteres de la cadena modelo */
MstrSetConstraint(MilStrContext, M_STRING_INDEX(0), 0, M_ANY, M_NULL );
MstrSetConstraint(MilStrContext, M_STRING_INDEX(0), 1, M_ANY, M_NULL );
MstrSetConstraint(MilStrContext, M_STRING_INDEX(0), 2, M_ANY, M_NULL );
MstrSetConstraint(MilStrContext, M_STRING_INDEX(0), 3, M_ANY, M_NULL );
MstrSetConstraint(MilStrContext, M_STRING_INDEX(0), 4, M_ANY, M_NULL );
MstrSetConstraint(MilStrContext, M_STRING_INDEX(0), 5, M_ANY, M_NULL );
MstrSetConstraint(MilStrContext, M_STRING_INDEX(0), 6, M_ANY, M_NULL );
MstrSetConstraint(MilStrContext, M_STRING_INDEX(0), 7, M_ANY, M_NULL );

```

Figura 5.10. Extracto de código en el que se establece el formato de la cadena que se espera identificar en la imagen de entrada.

5.1.1.6. Análisis de la imagen

Una vez definidos configurados los parámetros de la cadena de entrada, se procede a resetear el display para poder realizar la lectura de la nueva imagen que se cargará a continuación. Esto se consigue utilizando la función `MdispControl`. Con el fin de obtener la imagen digital capturada por la cámara en escala, se utiliza la función `MbufcopyColor`, ya que la función `MstrRead` y `MstrEditFont` no soportan como entrada una imagen de más de 8 bits. La función `MstrRead` asocia el buffer de memoria de la imagen con las funciones `MilStrContext` y `MilStrResult`. Además, se realiza una lectura del temporizador o reloj del sistema mediante la función `MappTimer` para obtener el tiempo empleado en el proceso (ver Figura 5.11).

```

/* Se elimina la imagen cargada en el display */
MdispControl(MilDisplay, M_OVERLAY_CLEAR, M_DEFAULT);

/* Se carga la captura de la cámara cargando
sólo una capa para obtener una imagen en escala de grises */

MbufCopyColor(MilImageDig, MilImage, M_RED);
MstrPreprocess(MilStrContext, M_DEFAULT);
MstrRead(MilStrContext, MilImage, MilStrResult);

/* Se resetea el reloj */

MappTimer(M_TIMER_RESET+M_SYNCHRONOUS, M_NULL);

MstrRead(MilStrContext, MilImage, MilStrResult);

/* se lee el reloj para obtener la duracion del proceso */
MappTimer(M_TIMER_READ+M_SYNCHRONOUS, &Time);

```

Figura 5.11. Extracto de código en el que se resetea el display y se obtiene la duración del proceso.

Mediante la función `MstrGetResult` se guarda en la variable `NumberOfStringRead` el número de cadenas leídas por el programa y con `MgraColor` y `MstrDraw` se dibuja sobre la imagen capturada los caracteres reconocidos y el cuadrado contenedor de dicha matrícula (ver Figura 5.12). Si el número de cadenas leídas es menor que uno, el programa indicará un error de lectura.

```

/*Obtiene el numero de cadenas leídas y lo muestra*/
MstrGetResult(MilStrResult, M_GENERAL, M_STRING_NUMBER+M_TYPE_LONG, &NumberOfStringRead);
if( NumberOfStringRead >= 1)
{
printf("Matricula leida en:(%.2lf ms)\n\n", Time*1000);

/* Se resaltan los caracteres reconocidos */

MgraColor(M_DEFAULT, M_COLOR_BLUE);
MstrDraw(M_DEFAULT, MilStrResult, MilOverlayImage, M_DRAW_STRING, M_ALL, M_NULL, M_DEFAULT);
MgraColor(M_DEFAULT, M_COLOR_GREEN);
MstrDraw(M_DEFAULT, MilStrResult, MilOverlayImage, M_DRAW_STRING_BOX, M_ALL, M_NULL, M_DEFAULT);

/* Saca por pantalla la matricula leída. */

printf(" Matricula: \n" );
printf(" -----\n" );
MstrGetResult(MilStrResult, 0, M_STRING, StringResult);
MstrGetResult(MilStrResult, 0, M_STRING_SCORE, &Score);
printf(" %s %.1lf\n", StringResult, Score );
}
else
{

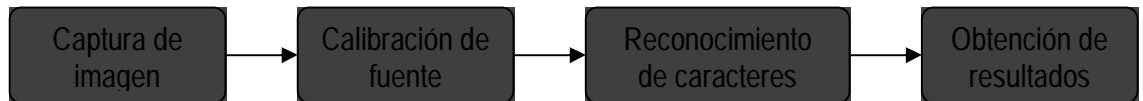
```

Figura 5.12. Extracto de código en el que se muestra, sobre-impressionada en la imagen original, la cadena de caracteres identificada en la matrícula.

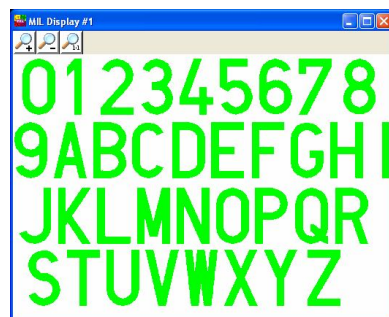
5.1.1.7. Análisis de los resultados

Una vez reconocida la matrícula, la cadena identificada queda almacenada en la variable `StringResult` para, posteriormente, comunicársela a la base de datos.

El proceso viene definido por el siguiente esquema:



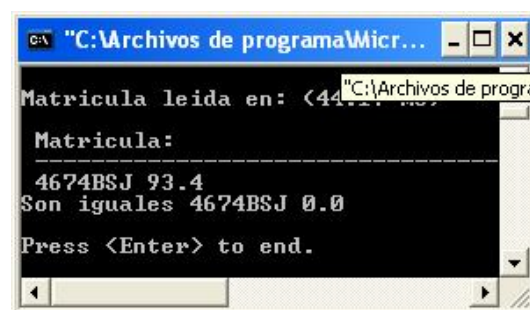
La aplicación captura una imagen de la matrícula del vehículo que intenta acceder cuando recibe una señal del autómata como que dicho vehículo se encuentra posicionado sobre un sensor de espira magnética situado bajo la barrera de acceso. El siguiente paso es calibrar la fuente de dicha matrícula mediante una imagen patrón con los caracteres de las matrículas españolas. Una vez calibrada la fuente, se realiza la lectura de la matrícula mediante reconocimiento de caracteres (OCR) y se obtienen los resultados de dicho reconocimiento como se muestra en las Figuras 5.13a, 5.13b y 5.13c.



a) Calibración patrón



b) Matrícula reconocida



c) Cadena obtenida con el número de matrícula

Figura 5.13. Capturas de pantalla de la aplicación de reconocimiento de matrículas implementada. (a) Patrón utilizado en para la calibración, (b) Imagen resultado en la que se muestran sobre-impressionados los caracteres identificados, (c) salida por pantalla indicando la cadera de caracteres identificada.

Se han llevado a cabo numerosas pruebas para medir el grado de robustez del sistema, teniendo en cuenta los siguientes parámetros:

- Ángulo de inclinación de la matrícula.
- Tamaño de los caracteres.
- Distancia de la matrícula a la cámara de visión.

Para comprobar el impacto de la inclinación de la matrícula en la imagen, debida a posibles desajustes o problemas de fijación de la cámara, se han realizado los siguientes ensayos:

1. Caso de estudio con una matrícula inclinada 330° (-30°).

Como se puede observar en la Figura 5.14, en la que la matrícula aparece inclinada -30°, los caracteres son reconocidos correctamente (aparecen resaltados sobre la imagen original). En este caso, el proceso de reconocimiento tardó 38.16 ms. Tanto la distancia de la cámara a la matrícula como la apertura de la lente para el ajuste de luminosidad se realizó manualmente. Uno de los problemas surgidos en estos ensayos fue calcular el rango de apertura del obturador de la cámara para controlar el paso de luz, ya que tanto con una apertura muy baja como con una apertura grande, el sistema era incapaz de reconocer las cadenas.

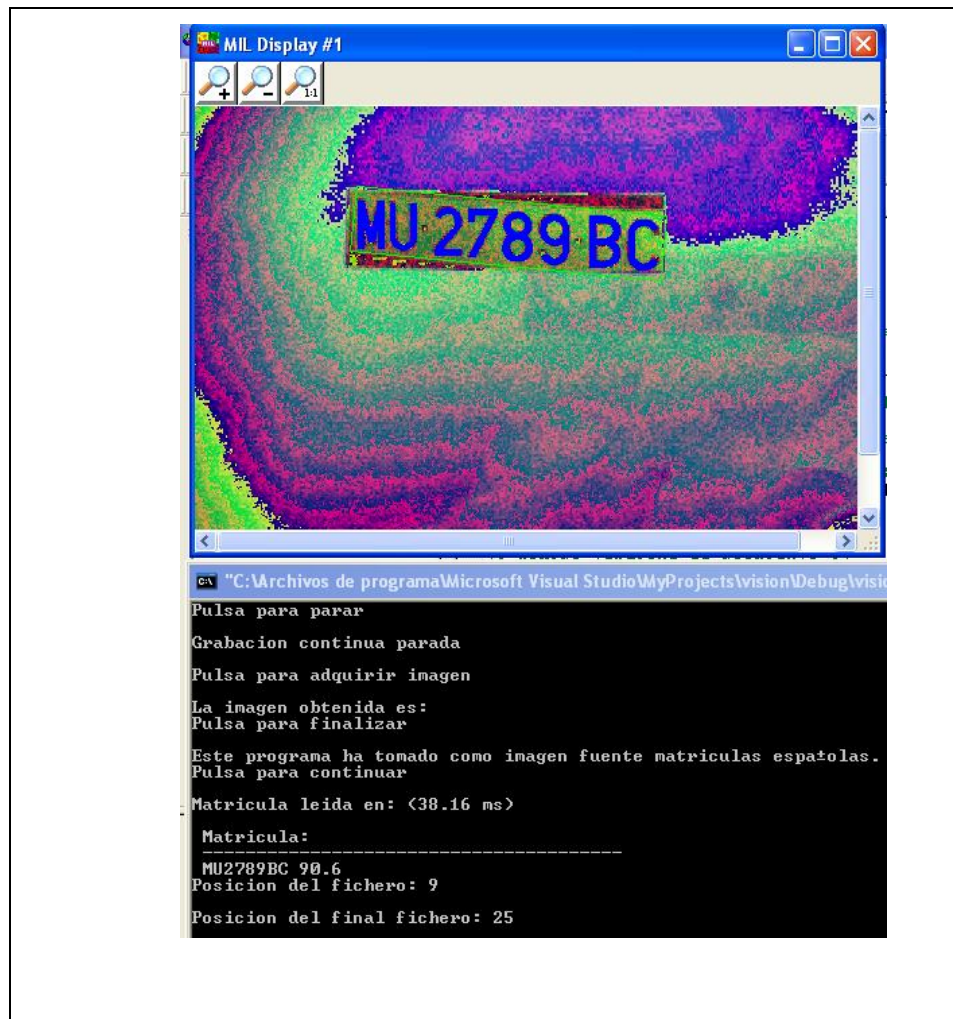


Figura 5.14. Caso de estudio: reconocimiento de los caracteres de una matrícula inclinada -30°.

2. Caso de estudio con una matrícula no inclinada.

Para este caso de estudio se utilizó una matrícula completamente alineada con la horizontal (no inclinada). Como en el caso anterior, los resultados de identificación de los caracteres de la matrícula fueron correctos. En este caso, el proceso tardó 47.86 ms; algo menos que en el ensayo anterior debido a una reducción de la luminosidad sobre el área donde se encuentra la matrícula (ver Figura 5.15).

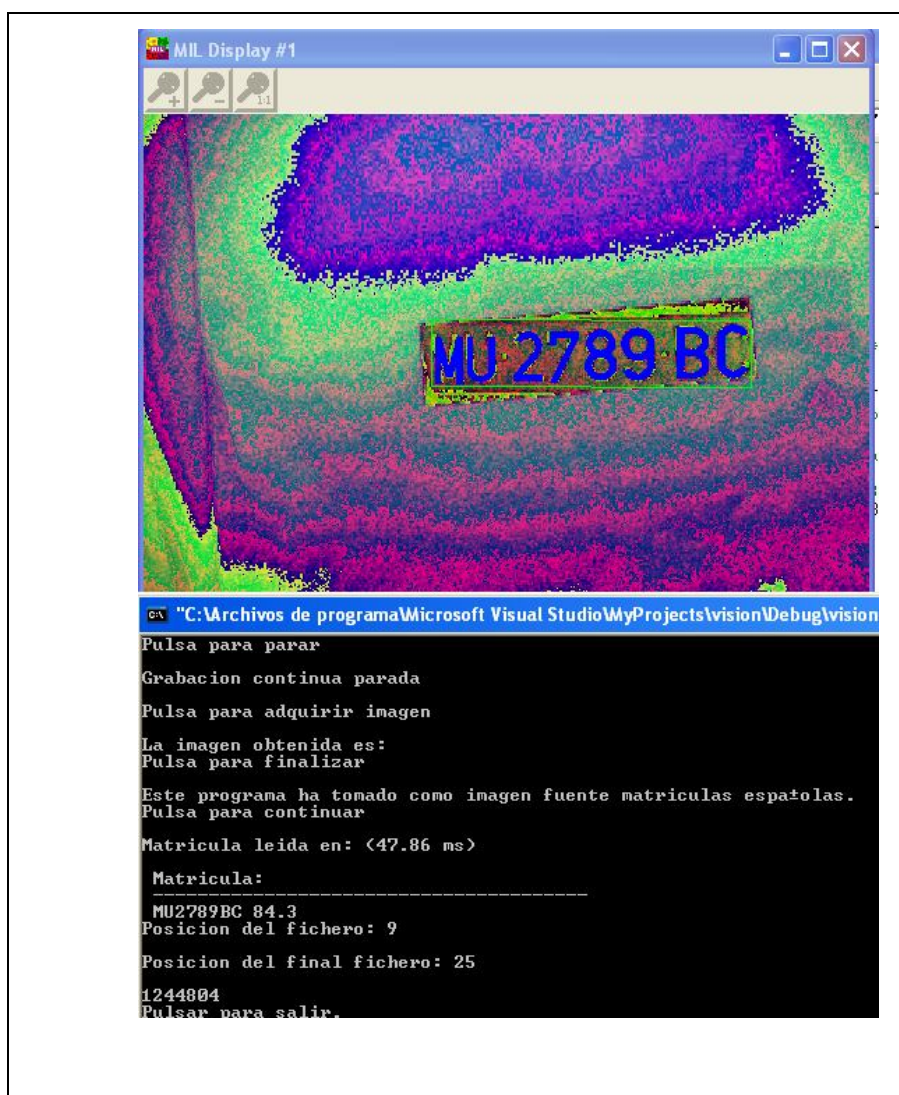


Figura 5.15. Caso de estudio: reconocimiento de los caracteres de una matrícula no inclinada.

También se llevaron a cabo algunos experimentos con distintos tamaños de caracteres. Estas variaciones suelen ocurrir cuando los vehículos se sitúan a distintas distancias de la cámara. A continuación se muestran los resultados obtenidos en varios ensayos.

1. Caso de estudio con una matrícula situada a 1 metro de la cámara.

Este estudio se realizó fijando una distancia aproximada de 1 metro desde la lente hasta la matrícula. La variación de distancia se realizó manualmente mediante el soporte de la cámara. Como se puede observar en la Figura 5.16, la matrícula tenía un ángulo de

inclinación de -40° y una iluminación baja. El resultado de este caso concluyó con la correcta lectura de la matrícula en un tiempo de 46.09 ms.

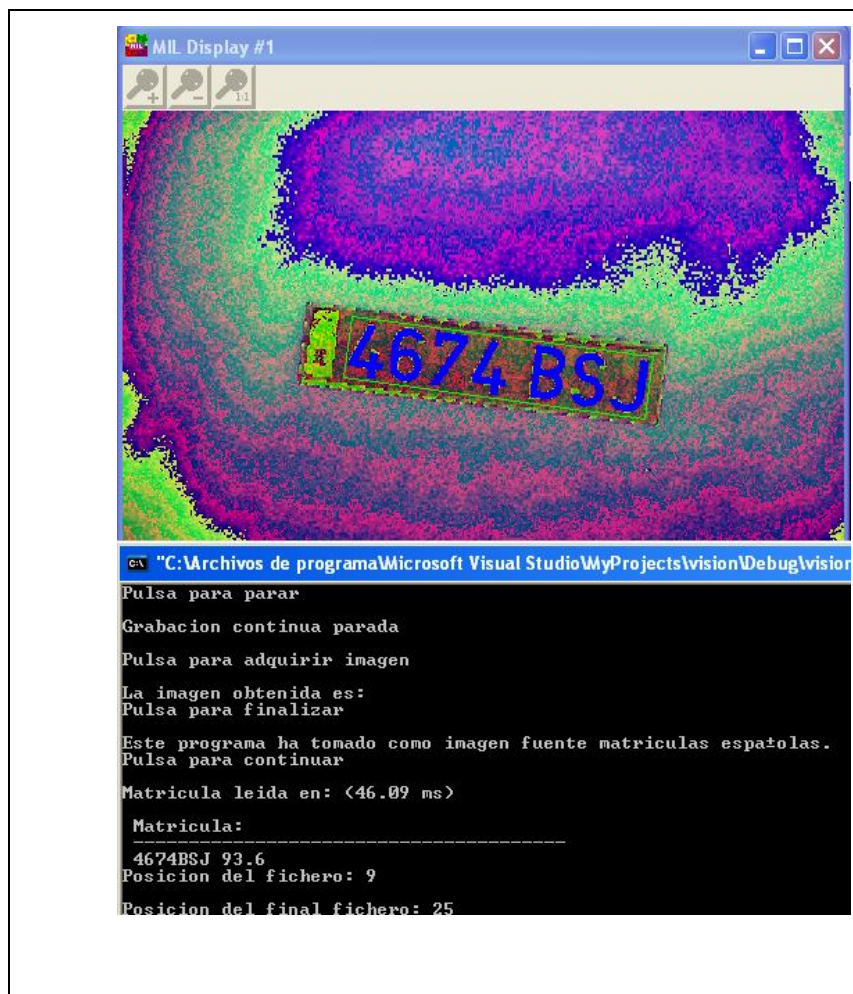


Figura 5.16. Caso de estudio: reconocimiento de los caracteres de una matrícula inclinada -40° y situada a 1 metro de distancia de la cámara.

2. Caso de estudio con una matrícula situada a 0.5 metros de la cámara.

Para comprobar el correcto funcionamiento de la aplicación al variar las condiciones de trabajo, se redujo a la mitad la distancia entre la cámara y la matrícula anterior, con lo que los caracteres que aparecen en la imagen tienen un tamaño considerablemente mayor. El resultado de este caso concluyó con la correcta lectura de la matrícula en un tiempo de 40.03 ms (ver Figura 5.17).



Figura 5.17. Caso de estudio: reconocimiento de una matrícula situada a 0.5 metros de la cámara.

Para finalizar el estudio, se realizaron dos ensayos más. Ambos casos corresponden a un error de lectura de la cadena introducida. En el primer caso (ver Figura 5.18), se puede observar que el error viene marcado por una baja luminosidad en la sala acompañada de un mal enfoque de la lente. En el segundo caso (ver Figura 5.19), el error se debe a un ángulo de inclinación fuera del rango de trabajo de la aplicación. En ambos casos, se imprime por pantalla "Matrícula no leída" y la cadena `StringResult` queda vacía. En el caso de que la aplicación no reconociese la matrícula, se procedería igual que en la situación de matrícula leída pero no perteneciente a la base de datos, con lo que se mandaría la petición de expedición de ticket de entrada.

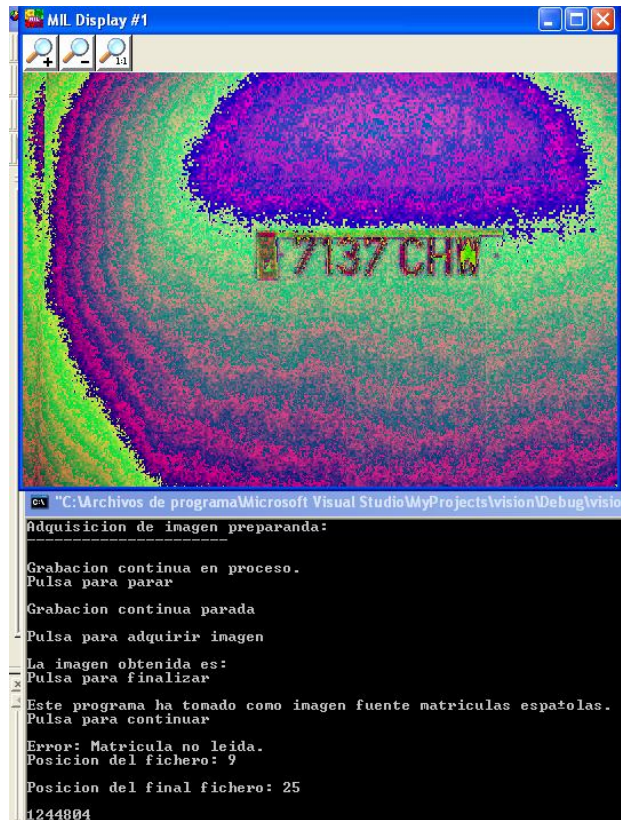


Figura 5.18. Error en el reconocimiento de una matrícula capturada con baja luminosidad y mal enfoque.

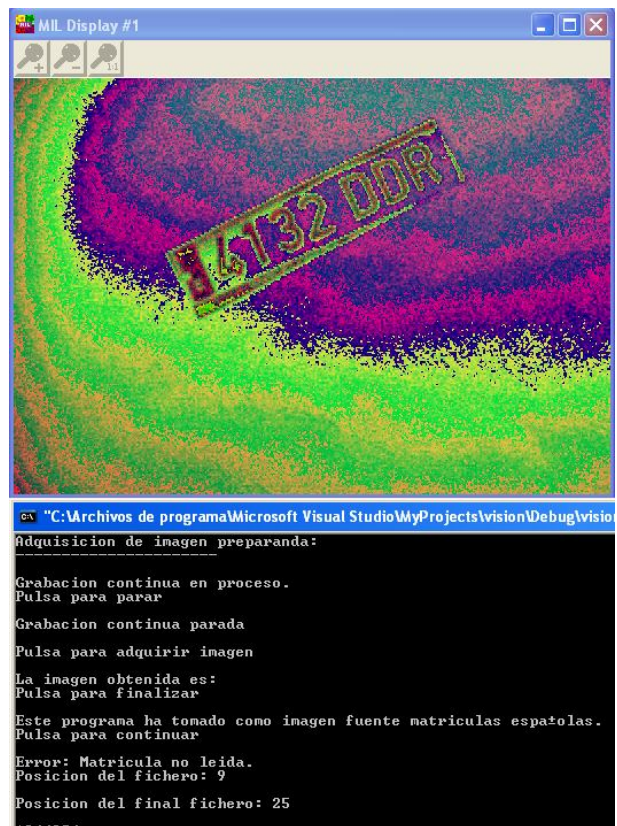


Figura 5.19. Error en el reconocimiento de una matrícula con un grado de inclinación excesivo.

5. 2. Migración de la aplicación de MIL a ActiveMil.

En apartados anteriores se ha detallado la aplicación desarrollada utilizando el lenguaje C y las librerías MIL. Sin embargo, como ya se comentó antes, esta aplicación ha sido re-programada en C++ con el fin de facilitar su integración con el resto de aplicaciones implementadas para la gestión del parking.

La aplicación C++ desarrollada es muy similar a la anterior programada en C, pero existen diferencias obvias debidas a que C es un lenguaje estructurado, mientras que C++ es orientado a objetos. A continuación se señalan los cambios más significativos realizados en la nueva aplicación.

Programación en C	Programación en C++	Tarea
MbufRestore(IMAGE_FILE_DEFINITION, MilSystem, &MillImage);	CString strActiveMILImagesPath = static_cast<LPCTSTR>(m_Application1->ActiveMILImagesPath); CString strActiveMILImageFile = strActiveMILImagesPath + IMAGE_FILE_DEFINITION; m_Image1->Load(_bstr_t(strActiveMILImageFile), FALSE);	Se realiza la carga de la imagen patrón en el buffer de memoria
MstrEditFont(MilStrContext, M_FONT_INDEX(0), M_CHAR_ADD, M_USER_DEFINED + M_FOREGROUND_BLACK, MillImage, TEXT_DEFINITION, M_NULL); MstrDraw(M_DEFAULT, MilStrContext, MilOverlayImage, M_DRAW_CHAR, M_FONT_INDEX(0), M_NULL, M_ORIGINAL);	IStringReaderFontCharactersPtr CharactersPtr = m_StringReader->Fonts->Item(1)->Characters; CharactersPtr->AddFromImage(m_Image1, strForegroundBlack, TEXT_DEFINITION); CharactersPtr->Draw(m_GraphicContext1, static_cast<StrDrawCharactersOperationConstants>(strDrawCharacters + strCharactersOriginalPosition));	Se calibra la cadena de la imagen patrón y se dibujan sobre los caracteres reconocidos
MdigGrabContinuous(MilDigitizer, MillImageDig); MdigHalt(MilDigitizer); MdigGrab(MilDigitizer, MillImageDig);	m_Digitizer->GrabContinuous(); m_Digitizer->MDigHalt(); m_Digitizer->MDigGrab();	Se captura la imagen de la matrícula
MstrPreprocess(MilStrContext, M_DEFAULT); MstrRead(MilStrContext, MillImage, MilStrResult); MstrRead(MilStrContext, MillImage, MilStrResult); MstrGetResult(MilStrResult, M_GENERAL, M_STRING_NUMBER + M_TYPE_LONG, &NumberOfStringRead);	CString strActiveMILImagesPath = static_cast<LPCTSTR>(m_Application1->ActiveMILImagesPath); CString strActiveMILImageFile = strActiveMILImagesPath + m_Image1;	Se realiza el reconocimiento de caracteres de la imagen capturada

La aplicación esta programada en Microsoft Visual Studio.Net 2003 y se usan funciones obtenidas de las librerías de Matrox MIL 8.0. (Ver Figura 5.20)

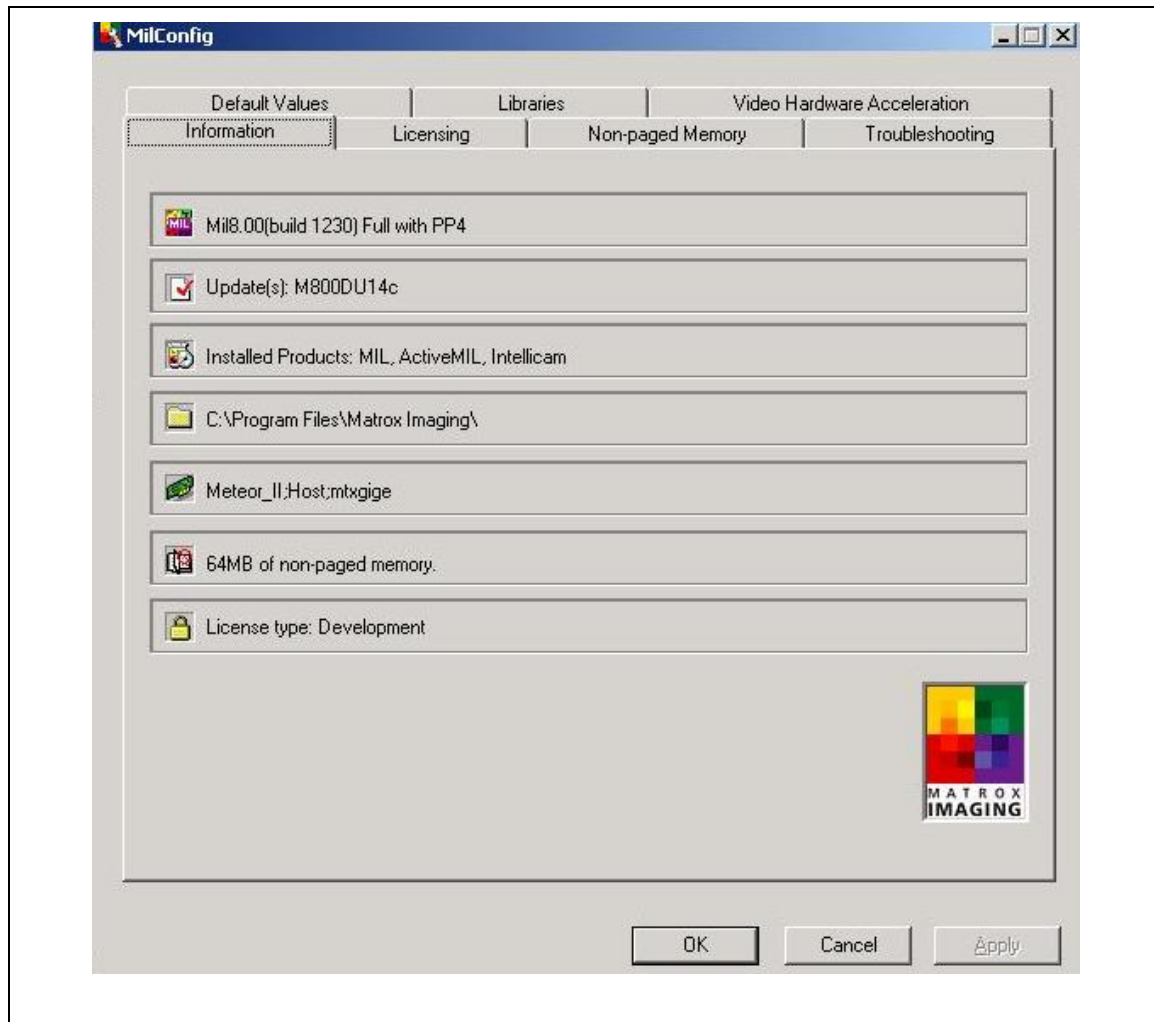


Figura 5.20. Configuración librerías MIL

Para la realización del programa en C++ se utilizó la herramienta ActiveMIL, que consiste en un conjunto de controles ActiveX que se basan las librerías MIL (Matrox Imaging Library) y esta diseñada para el desarrollo rápido de aplicaciones. (Ver Figura 5.21)

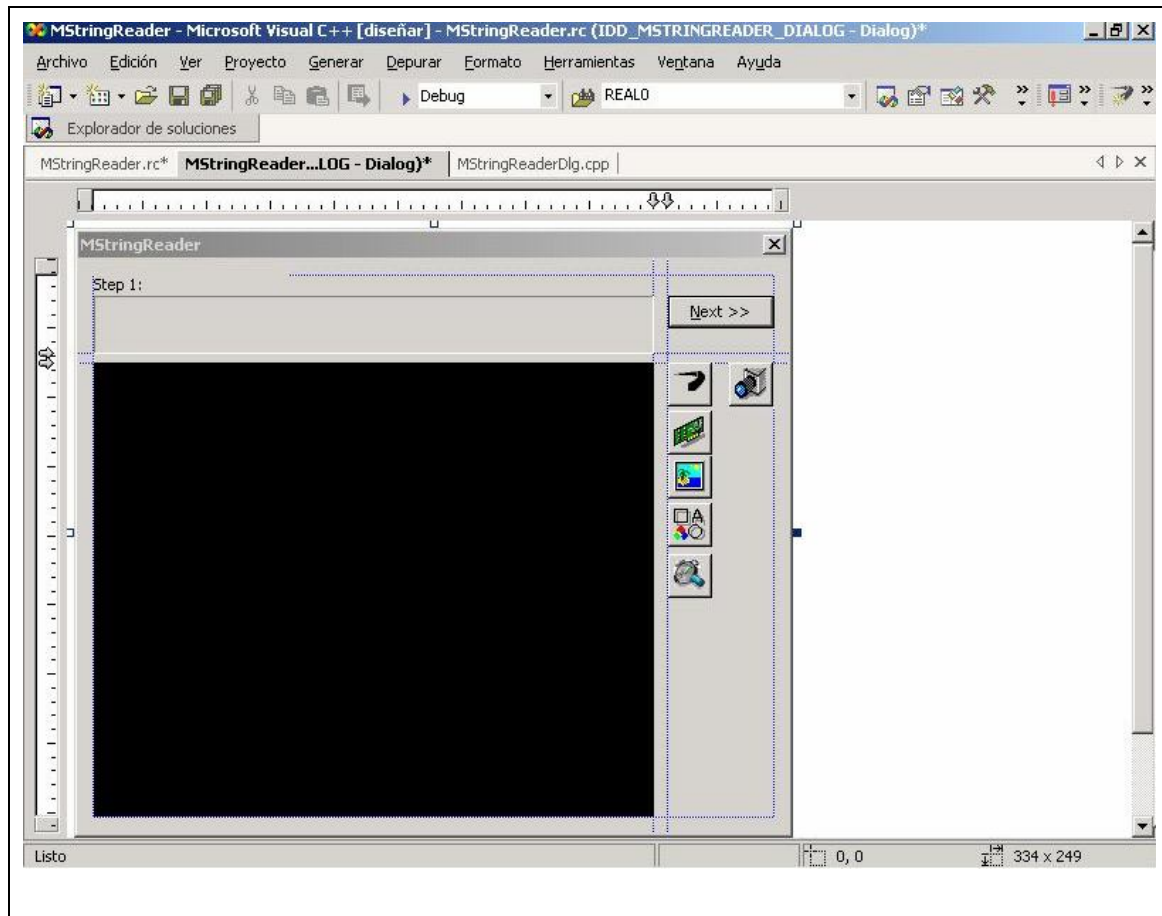


Figura 5.21. Ejemplo de utilización de ActiveMIL

CAPÍTULO 6

CONCLUSIONES Y FUTUROS TRABAJOS

Se ha implementado una aplicación de reconocimiento de matrículas, basada en técnicas de Visión Artificial, para controlar el acceso a un parking semi-público. Se han utilizado herramientas específicas para el desarrollo de este tipo de aplicaciones, tales como las librerías MIL 8.0 o el entorno de desarrollo Microsoft Visual Studio. Éste último permite a crear aplicaciones, sitios y aplicaciones web, así como servicios web en cualquier entorno que soporte la plataforma .NET. Así se pueden crear aplicaciones que se intercomunique entre estaciones de trabajo, páginas web y dispositivos móviles.

La mayoría de los parking que se construyen en la actualidad ofrecen funcionalidades de reconocimiento de matrículas, ya que esto permite un mayor control de acceso y un ahorro en tiempo y comodidad para los clientes residentes.

En futuros proyectos, se podría mejorar la aplicación introduciendo algoritmos que permitan la eliminación de suciedad de la lente mediante tratamiento de imágenes, la calibración de matrículas con ángulos de inclinación en 3D, o incluso la integración de la aplicación en un sistema de circuito cerrado de vigilancia.

ANEXO 1

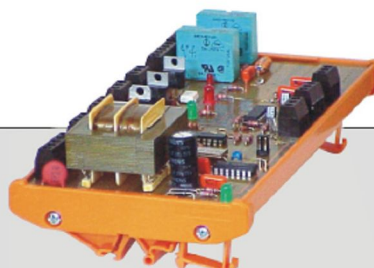
ESPECIFICACIONES TÉCNICAS

1.1. Características del detector de vehículos por espiras



DVPE2

DETECTOR VEHICULAR POR ESPIRA



ENTRADAS

De osciladores

Cantidad: 2
Distancia máxima a oscilador: 50 m
Digitales
Cantidad: 2
Tipo: contacto seco
Tensión nom: 24 Vcc, (interna).

SALIDAS

Relés

Cantidad: 2
Capacidad: 5 A @ 220 Vca
Contactos secos: C, NA, NC.
Led indicador
Triac
Cantidad: 4
Capacidad: 300 W @ 220 Vca
Optoacoplados, disparo en cruce por cero.
Salidas para lámparas
Leds indicadores: 2 verdes, 2 rojos.

ALIMENTACION

Tensión: 220 Vca - 15%, + 10%, 50 Hz
Consumo (sin lámparas): 8 VA
Led indicador de alimentación

ESPIRAS

Cantidad de vueltas: 3
Profundidad de enterrado: 3 a 15 cm
Dimensiones: 1,80 m por 1,80 m
Distancia máxima a oscilador: 2 m

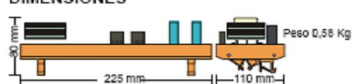
MODELOS

DVPE2S modelo simple, una espira.
DVPE2D modelo doble, dos espiras.

OPCIONAL

Puerta serial RS-485

DIMENSIONES



OSCILADOR



FABRICADO EN URUGUAY

FUNCION

Indicar la presencia de un vehículo que circula sobre la espira.

CARACTERISTICAS

Detector vehicular doble con salida por relé y directa para lámparas por triac.

El detector se compone de tres partes, las espiras que se entierran en el pavimento, los osciladores y el circuito lógico.

FUNCIONAMIENTO

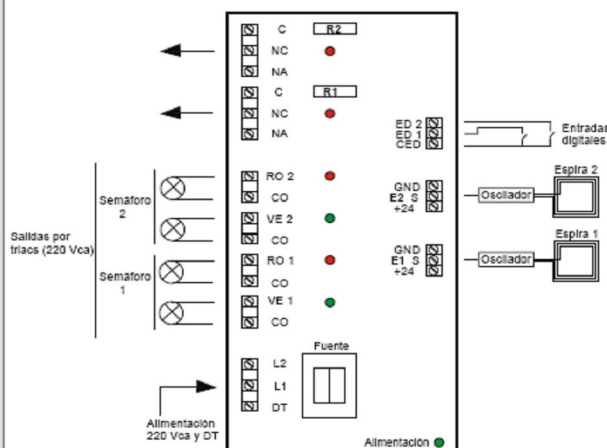
Cuando un vehículo circula sobre la espira oscilante enterrada en el pavimento modifica la frecuencia, el cambio es detectado por el circuito lógico que genera la señal de salida en relé y triac.

Cuando se implementan las dos espiras las señales de salida indican el sentido de avance del vehículo

APLICACIONES TÍPICAS

Detector para semáforo
Barreras de acceso a estacionamiento
Contador vehicular
Rampas de garages

DIAGRAMA



Tel.: +598 2 622 0651
Fax: +598 2 622 2048
info@controles.com
www.controles.com

CONTROLES S.A. Av. Rivera 3314
C.P. 11300 - Montevideo - URUGUAY

1.2. Características del detector inductivo de vehículos

CARACTERÍSTICAS TÉCNICAS

- Microcontrolador
- 1 ó 2 canales.
- Fuente de alimentación conmutada.
- 4 niveles de sensibilidad seleccionables.
- 4 frecuencias seleccionables.
- Salidas opto aisladas ó a Relé.
- Salida con seguridad ante falla.
- Multiplexado de canales.
- Diagnóstico de fallas avanzado.
- Salto automático de sensibilidad.
- Detección de sentido de circulación.
- Sintonía automática.
- Salida por presencia o por pulso.
- Seguimiento de fluctuaciones ambientales.



Funcionamiento del Sistema

El usuario ingresa cuando aproxima la tarjeta al lector. Este decodifica la información de la tarjeta (el número que tiene asignado) y se comunica con el Servidor de datos. Estando la tarjeta habilitada, la barrera se levanta y permite el paso del vehículo. Una vez que éste pasa por el detector inductivo de vehículos, baja y el sistema queda habilitado para recibir otro usuario. Una vez ingresado, el evento es registrado en la base de datos. El programa ACCESO de VITEC presenta en pantalla, en forma gráfica, la información de lo que acontece en el acceso, informando el nombre del usuario asignado a esa tarjeta y el estado de los sensores y equipos.

1.3. Características sensor óptico reflectivo



FBOTSTR

FUENTE Y RELE PARA SENSOR
OPTICO REFLECTIVO



ENTRADAS

Alimentación: 220 Vca, 50 Hz.
Bornes: A1, A2.
Consumo menor que 3 Va.
Señal (S): 18 a 24 Vcc.

SALIDAS

Alimentación para sensores: + 24 Vcc.
Indicadores luminosos: rojo y verde.
Relé inversor: contactos: NC, C, NA.
Capacidad de los contactos: 5A@220 Vca.

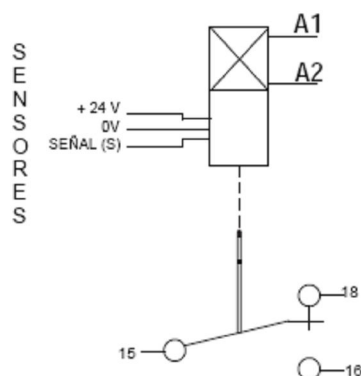
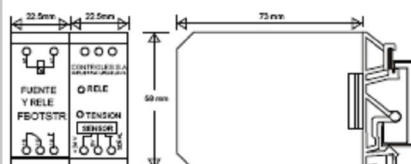
FUNCION

Alimentar sensores ópticos reflectivos modelo BOTSTR y reproducir la señal mediante un relé inversor.

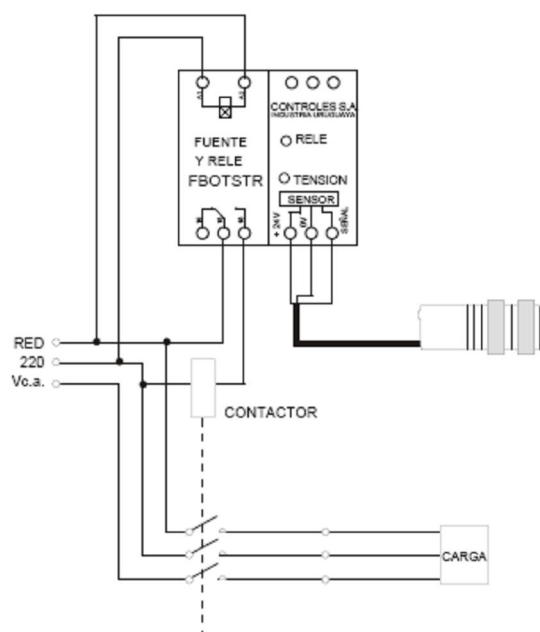
OPERACION

El indicador verde encendido indica equipo funcionando.
El indicador rojo esta encendido y el relé accionado, cuando la señal de entrada está activada.
Ver modos de operación en el folleto de sensor óptico reflectivo.

Ejemplo de conexión "ve retorno"



FABRICADO EN URUGUAY



CONTROLES S.A. Av. Rivera 3314
C.P. 11300 - Montevideo - URUGUAY

Tel.: +598 2 622 0651
Fax: +598 2 622 2048
info@controles.com
www.controles.com

1.4. Características de las barreras infrarrojas



BOTST

BARRERA INFRARROJA



PRINCIPIO DE FUNCIONAMIENTO
Haz de luz infrarrojo pulsado y sincronizado.
Distancia máxima: 8 m
Tiempo de reacción: 0.5 s

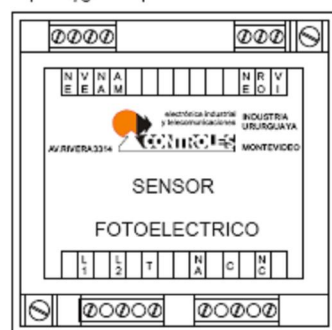
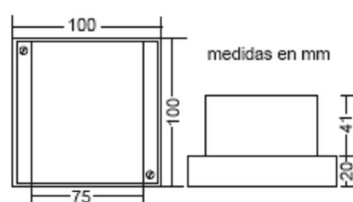
SALIDA
Relé inversor: NA, C, NC
Capacidad contactos: 5A @ 220 Vca
Señalización del relé: led rojo

ALIMENTACION
220 Vca $\pm 10\%$, L1 L2
Consumo: menor que 3 VA
Borne de tierra: T
Sensores: 3 m de cable de extensión.

MODELOS
Sin temporización: BOTST
Con temporización: BOTCT

PESO
BOTST c / sensores: 0,71 Kg.

GABINETE



Rev 1.0

DS13/06

FABRICADO EN URUGUAY

FUNCION

Detectar la interrupción del haz infrarrojo y conmutar el relé.

OPERACION

Con el haz de infrarrojo establecido el led enciende y el relé cierra.
Cuando el haz se interrumpe el led se apaga y el relé se abre.
En la versión temporizada el relé abre cuando el haz es interrumpido por un tiempo mayor que el valor de la temporización.

CARACTERISTICAS

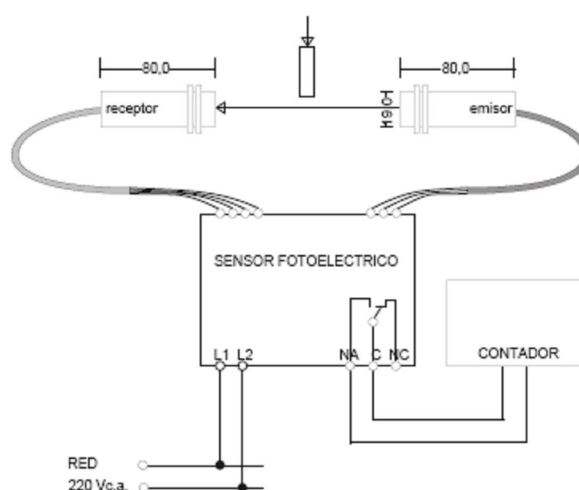
- Elevada inmunidad a las interferencias luminosas.
- Haz pulsado con ventana de recepción sincronizada.
- Invisible al ojo humano
- Permite el empleo en el modo reflectivo.
- Fácil enfoque.

Se compone de tres elementos, el control, el emisor del haz y el receptor del haz.

APLICACIONES

Contador de objetos en líneas de producción, vigilancia de pasajes, puertas de ascensores, llenado de recipientes, etc.

EJEMPLO DE APLICACION



CONTROLES S.A. Av. Rivera 3314
C.P. 11300 - Montevideo - URUGUAY

Tel.: +598 2 622 0651
Fax: +598 2 622 2048
info@controles.com
www.controles.com

1.5. Características de los sensores ultrasónicos

Sensor ultrasónico Sipark SSD (Siemens)

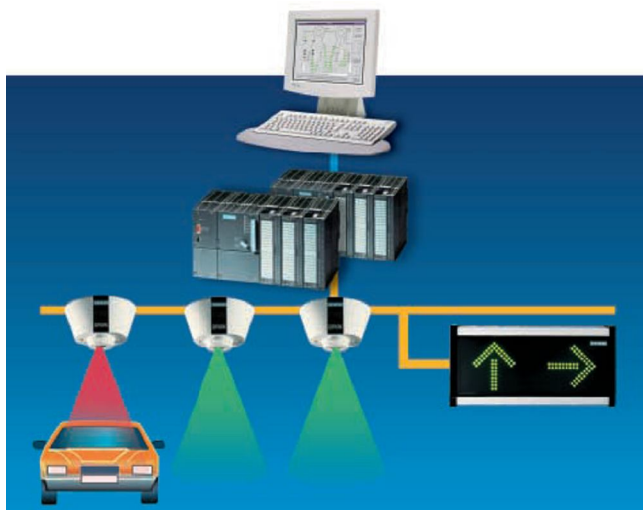
El sensor ultrasónico de SIPARK SSD se instala en el techo por encima de cada una de las plazas individuales y detecta con una precisión del 100% si la plaza está libre u ocupada. Esto permite al operador conocer en tiempo real cuántas y, en especial, cuáles de sus plazas de aparcamiento están ocupadas o libres en cada, a la vez que el conductor puede ver desde lejos donde se encuentran las plazas libres, ya que el sensor de ocupación lo indica mediante luz roja o verde.



El sensor ultrasónico detecta con absoluta seguridad si una plaza de aparcamiento está libre u ocupada (incluso por motocicletas)

Normalmente la altura de instalación permite únicamente el aparcamiento de turismos. Pero también existe una solución para la detección de plazas de aparcamiento para camiones: un sensor de techo con altura de detección especialmente elevada (5 metros). Si se requiere una detección altamente fiable de plazas libres, puede utilizarse un sensor frontal que analiza la plaza desde la parte delantera en lugar de desde el techo.

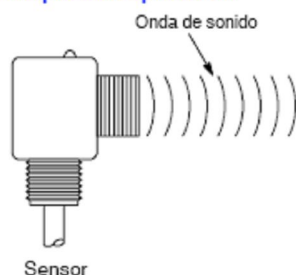
Para el usuario del aparcamiento, el sistema resulta ante todo agradable y práctico: lo conduce con seguridad y rapidez hasta el lugar donde podrá aparcar el coche. Pero el operador significa mucho más: ocupación optimizada de las plazas, reducción de emisiones, liberación flexible de plazas normales y especiales, un gran número de evaluaciones estadísticas...



Sensores ultrasónicos

Definiciones y terminología técnicas

Principios de operación



Los sensores ultrasónicos funcionan emitiendo y recibiendo ondas de sonido de alta frecuencia. La frecuencia normalmente está en el orden de los 200 kHz, por lo que es demasiado alta para que el oído humano la perciba.

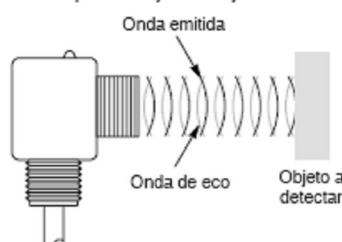
Modos de operación

Hay dos modos básicos de operación: modo opuesto y modo difuso (de eco).

En el modo opuesto, un sensor emite la onda de sonido y otro, que está montado al lado opuesto del emisor, recibe la onda de sonido.

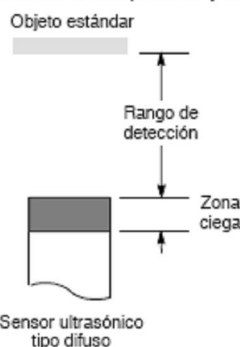


En el modo difuso, el mismo sensor emite la onda de sonido y luego busca el eco que un objeto refleja.



Rango de detección

El rango de detección es la distancia dentro de la cual el sensor ultrasónico detectará un objeto sometido a fluctuaciones de temperatura y voltaje.



Zona ciega

Los sensores ultrasónicos tienen una zona ciega inherente que se localiza en la cara de detección. El tamaño de la zona ciega depende de la frecuencia del transductor. Los objetos que se sitúan dentro del punto ciego no se podrán detectar de manera confiable.

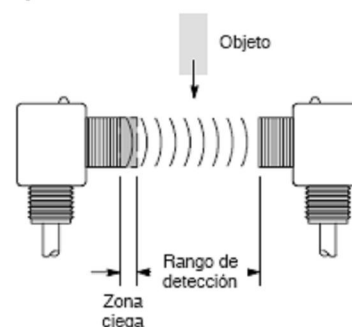
Consideraciones respecto al objeto

Se deben considerar ciertas características de los objetos cuando se usen sensores ultrasónicos. Ellas incluyen la forma, material, temperatura, tamaño y posición del objeto.

Los materiales suaves como las telas y el caucho de espuma son difíciles de detectar por medio de la tecnología ultrasónica difusa debido a que no reflejan el sonido.

El objeto estándar para un sensor ultrasónico de tipo difuso está establecido por el estándar IEC 60947-5-2 de la Comisión electrotécnica internacional. El objeto estándar es de forma cuadrada, con un espesor de 1 mm y está hecho de metal con acabado laminado. El tamaño del objeto está en función del rango de detección.

No se ha establecido un estándar para los sensores ultrasónicos de modo opuesto.



Los objetos estándares se usan para establecer los parámetros de funcionamiento de los sensores. El usuario debe tener en cuenta las diferencias de rendimiento producidas por los objetos no estándares.

Apripark Detector (Aprimatic)



Es un sistema de gestión automática de la ocupación del parking, proporciona al usuario la información necesaria para localizar rápidamente las plazas libres del aparcamiento. Este sistema permite reducir las retenciones en las vías de acceso al parking, al tiempo que ofrece al usuario mayor comodidad y rapidez a la hora de estacionar su vehículo.

Existen dos modos de funcionamiento:

Modo Autónomo

Cada sensor ofrece información sobre la ocupación de la plaza en la que está situado mediante indicador luminoso rojo/verde.

El color rojo indica que la plaza está ocupada, y el color verde que está disponible.

Modo Centralizado

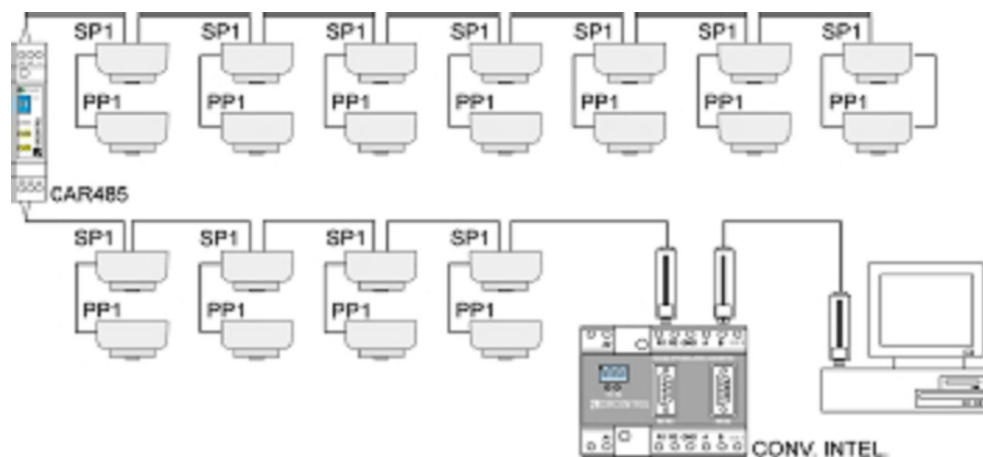
Los sensores en este tipo de instalación están conectados a un sistema que presenta la información del estado de ocupación de cada plaza en el puesto de control.

Equipo de detección de ocupación de plazas mediante ultrasonidos (Circontrol)

El sistema Circontrol de gestión de plazas de parking se basa en la detección de presencia de vehículo en cada plaza mediante un equipo de detección de ocupación de plaza mediante ultrasonidos (PPI).

La conexión de los diferentes sensores PPI mediante bus y de éste con el sistema informático facilita la información en tiempo real de la situación de cada plaza.

Esquema de instalación:



Elementos del sistema:

SP1:

Sensor de ultrasonidos para detectar la presencia de un vehículo en la plaza de aparcamiento
 Incorpora indicador luminoso del estado de la plaza
 Alimentación: 12 V c.c.
 Consumo: 120 mA
 Comunicaciones: RS-485
 Conexión para indicador luminoso externo PP1
 Rango de detección: de 30 cm a 1,70 m (otros rangos bajo demanda)



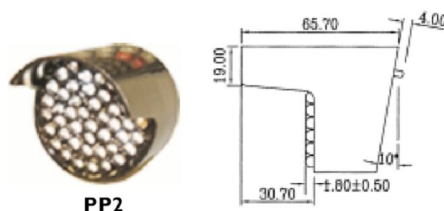
SP1

PP1:

Indicador luminoso del estado de la plaza
 Alimentación: 12 V c.c. (Suministrada por sensor SP1)
 Consumo: 50 mA
 Conexión a detector SP1

PP2:

PP2/24. Indicador luminoso estado de plaza
 Alimentación: 24 V c.c.
 20 diodos rojos, 30 diodos verdes
 Iv (mcd) 6000



1.6. Barrera vehículos RAP6

CARACTERÍSTICAS TÉCNICAS:

- Tensión de línea: 230 Vac.
- Alimentación motor: 24 Vdc
- Corriente máx. Absorbida (24Vdc): 1 A
- Potencia máx. absorbida: 200 VA
- Par nominal: 280 Nm
- Tiempo de abertura: 7-8 seg.
- Movimiento: irreversible
- Angulo máx. de rotación: 90°
- Temperatura de funcionamiento: -10 a +70°C
- Grasa motor: TS-10
- Intermitencia trabajo: 50%
- Longitud máx. asta: 6,25 m.
- Peso: 55 Kg

Complementos accesorios:

- Estructura móvil tipo valla.
- Pié articulado de apoyo.
- Asta con perfil rectangular L = m 6.25
- Mástil de perfil redondo L=6,25 m
- Apoyo fijo para asta

Funcionamiento del sistema

RAP-6 es una barrera electromecánica que se emplea en las entradas a aparcamientos, garajes, establecimientos, organismos públicos, hospitales, edificios residenciales, etc. Está constituida por un armario, en cuyo interior hay montado un motor reductor y una central

electrónica de mando. El movimiento es irreversible con la posibilidad de desbloqueo manual.

1.7. Barrera vehículos electromecánica GARD 4

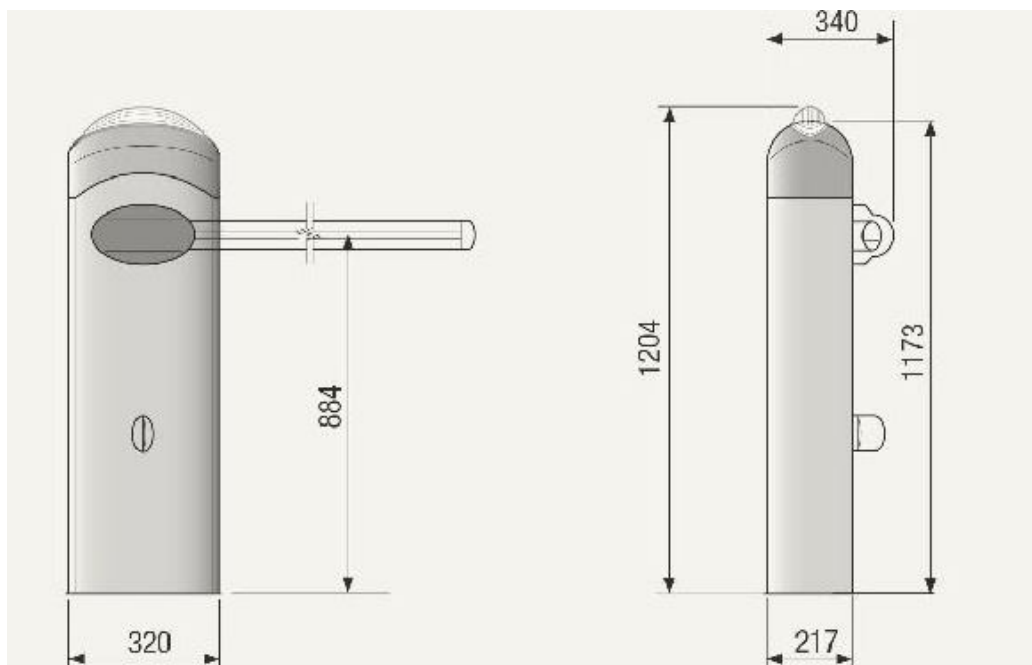
CARACTERÍSTICAS TÉCNICAS:

- Grado de protección: IP54.
- Alimentación (50/60Hz): 230V AC
- Alimentación Motor 230V AC (50/60Hz) o 24V DC.
- Potencia 250W-300W.
- Relación de reducción: 1/202.
- Empuje: 220Nm
- Tiempo de Apertura (90°) 4seg.



Complementos accesorios:

- Central de accesos on-line OPEN IP4.
- Mando emisor GO-2.
- Mando emisor GO-4.
- Barrera manual ACR-051.
- Mando emisor ACTING.
- Emisor manos libres GO FREE.



1.8. Anexo Barrera vehículos RAP-

CARACTERÍSTICAS TÉCNICAS:

- Tensión de línea: 230 Vac.
- Alimentación motor: 24 Vdc
- Corriente máx. Absorbida (24Vdc): 6 A
- Potencia máx. absorbida: 180 VA
- Par nominal: 260 Nm
- Tiempo de abertura: 5-6 seg.
- Movimiento: irreversible
- Angulo máx. de rotación: 90°
- Temperatura de funcionamiento: -10 a +70°C
- Grasa motor: TS-10
- Intermitencia trabajo: 50%
- Longitud máx. asta: 4,25 m.
- Peso: 43 Kg



Complementos accesorios:

- Estructura móvil tipo valla.
- Pié articulado de apoyo
- Asta con perfil rectangular L = m 4.25
- Apoyo fijo para asta

1.9. Barrera rápida ACGRM1

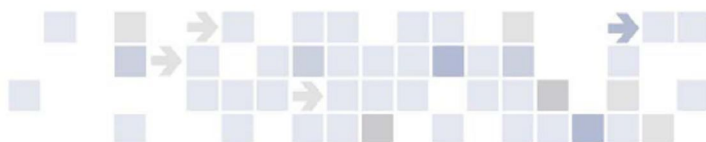
CARACTERÍSTICAS TÉCNICAS:

- Motor 230V AC / 550W
- Longitud mástil 3m.
- Tiempo apertura 1seg.
- Dimensiones 350x300x1070mm
- Peso 68Kg.
- Cierre automático.
- Ciclo de trabajo 100%.
- Margen de temperatura -30°C hasta +80°C



1.10. Anexo barrera BAR-XX masterasp

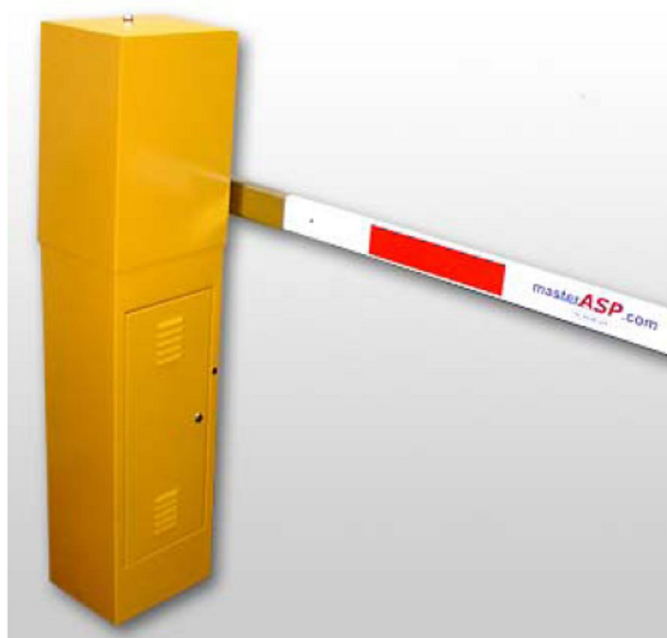
masterASP
www.masterasp.com



Características

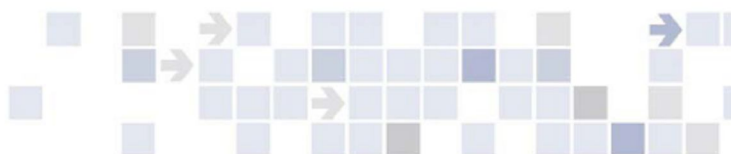
- Alta velocidad de apertura / cierre, hasta 1 segundo
- Chasis metálico resistente para intemperie
- Brazo de aluminio de sección rectangular de hasta 4 metros
- Totalmente electrónica, sin relés ni micros finales de carrera
- Uso intensivo, 2 millones de maniobras garantizadas
- Fácil montaje utilizando un anclaje metálico
- Doble lazo inductivo y fotocélula de seguridad
- Detector y memoria de obstáculos
- Posicionamiento por control numérico, movimiento programable
- Interface bus CAN 2.0 integrada
- Control adicional por entradas /salidas y por radiomandos.
- Programable remotamente

BAR-XX Barrera Automática



Opciones

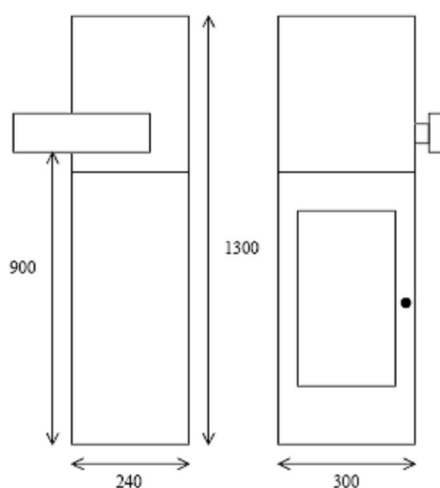
- Apertura automática en caso de fallo de alimentación
- Fotocélula de seguridad con soporte para el emisor
- Receptor de radio con varios mandos
- Semáforo rojo / verde
- Botonera externa abrir / cerrar / bloquear
- Mástil articulado
- Color personalizado bajo pedido (por defecto RAL 1004)



La barrera automática *masterASP BAR-XX* ha sido diseñada especialmente para uso intensivo. El movimiento del mástil se realiza mediante control numérico, sin necesidad de usar elementos mecánicos (muelles, contrapesos ni topes). Así se consigue una barrera muy robusta de hasta 5 millones de maniobras con un mantenimiento mínimo.

Otra característica diferencial de estas barreras es la interface de comunicaciones CAN 2.0. Esto permite un control absoluto de la barrera, diferentes configuraciones y la posibilidad de actualizar el firmware. Además, la barrera se puede gobernar por señales de contacto NA y por radiomandos.

Medidas



Especificaciones Eléctricas

Alimentación: monofásica 220V/50Hz

Consumo máximo: 1,6A (300W)

Potencia motor: 0,16KW, 680 rpm

Conexión:

- Interface CAN: RJ45 blindado
- 2 lazos inductivos
- Bornes para control y botonera
- Salida +12VDC

Chasis

Chasis fabricado en plancha de hierro galvanizado y posterior pintado al horno con doble capa en color amarillo RAL1004.

Dim: 30cm Ancho, 24cm Fondo, 130cm Alto

Peso: 55Kg

Puerta frontal con cerradura de seguridad

Capucha desmontable

Electrónica y Configuración

La placa de control contiene interruptores de test y configuración así como LEDs de estado. Conectando un LCD o a través del bus CAN se puede modificar el funcionamiento de la barrera.

Algunos parámetros son:

- Frecuencia y sensibilidad de los bucles
- Botones de los radiomandos
- Acción programable en caso de obstáculo
- Tiempos mínimos del cierre por lazo
- Movimiento del mástil (aceleración, par, velocidad, recorrido, etc.)
- Firmware actualizable remotamente

Temperatura de funcionamiento de -20°C hasta 70°C .

1.11. Pilona automática Dakota 500

CARACTERÍSTICAS TÉCNICAS:

- Tiempo de trabajo: 6seg. (8seg. con ralentización en final de carrera en subida y bajada).
- Velocidad de translación: 9cm/seg.
- Lubricación: Aceite Shell Tellus T 15
- Maniobra manual: baja espontáneamente en caso de falta de corriente.



DIMENSIONES

- Cilindro a enterrar: Tubo acero Ø273x h915 mm tratado en cataforesis
- Columna fuera del suelo: Ø210 x h500 mm en acero trefilado grosor 60/10
- Tratamiento y revestimiento: Cataforesis con capa retro reflectante certificada
- Visibilidad activa: sombrete con LED alta luminosidad
- Resistencia a la neblina salina: 700 horas
- Grado protección: IP67
- Carga dinámica admitida: 10.500 - 15.000 Joules
- Carga estática axial admitida:
 - con vástago subido máx. 800 daN
 - con vástago bajado máx. 4.0000 daN
- Cable de alimentación: 10m
- Peso embalado: 95 Kg.

1.12. Barreras Automáticas CT70-850-RF50-1000

Con la barrera CT70-850, Maps Informática Industrial añade un nuevo equipo, de fabricación propia, a la gama propia de productos para el control y gestión de tráfico de vehículos.

Es utilizada en todo tipo de aplicaciones donde se requiere controlar el tráfico intenso de vehículos como en:

- Autopistas
- Peajes
- Aparcamientos públicos y privados
- Gasolineras.
- Centros comerciales, etc.



DISEÑO



FABRICACIÓN



INSTALACIÓN



MANTENIMIENTO

Diseño

Esta barrera ha sido diseñada por nuestro departamento de I+D, aprovechando nuestra larga experiencia (desde 1970) en los sectores de la automatización industrial, y está preparada para trabajar 24 horas/días y 365 días/año, bajo las condiciones de trabajo más duras y adversas. Sus características de fabricación la hacen ideal para trabajar indistintamente a la intemperie o en recintos cerrados.

A la Vanguardia Tecnológica

La utilización de materiales de bajo mantenimiento, un diseño mecánico simple y efectivo, sin concesiones al azar, y microprocesadores de última tecnología, proporcionan a nuestras barreras unas prestaciones inigualables en cuanto a velocidad de apertura, bajo mantenimiento, y larga duración del equipo.

Flexibilidad

La Barrera automática admite mástiles de cualquier longitud hasta 5,7 metros y el modelo RF50, controlado por microprocesador admite diferentes configuraciones de velocidad de apertura/cierre.

Admite infinidad de accesorios con el fin de controlar el acceso de los vehículos de la forma más segura y ergonómica tanto para el usuario como para el propietario de la instalación. Accesorios fabricados por MAPS:

- Detectores de vehículos (inductivos, magnéticos, ultrasonidos, fotocélulas, etc.).
- Cortina fotoeléctrica para separar vehículos (contaje, clasificación, sentido de paso, número de ejes, altura, etc.).
- Control centralizado (cálculo de flujos, monitorización estado, bases de datos, etc.).
- Captura digital de matriculas: organizadas por fecha y hora.

Longitud de mástil

Se fabrican dos versiones que se diferencian en los aspectos de control y que indistintamente admiten mástiles de hasta 5'7 metros de longitud.

Velocidad

La velocidad de apertura y cierre de nuestras barreras baten todos los records. La CT70-850 realiza el movimiento en un tiempo de 1,5 segundos y las RF50-1000 lo realiza en tan solo 0,6 segundos (con mástiles de menos de 3 metros de longitud).

Precio / Calidad

Factores tales como el diseño mecánico simple a la vez que robusto, la utilización de materiales estándar de mercado, y un alto índice de satisfacción de nuestros clientes, nos permite poder ofrecer nuestras barreras a unos precios sin competencia. El empleo de materiales y tratamientos de protección como la cataforesis en el bastidor, el zincado de las piezas mecánicas y la utilización de componentes estándar de mercado de alta tecnología, son garantía de larga vida.

Seguridad

Todas las barreras automáticas fabricadas por MAPS, disponen como equipo de serie, de una célula fotoeléctrica, que asegura la inversión en el movimiento del mástil cuando un vehículo o persona se encuentra en la trayectoria del cierre.

Descripción del Equipo

- Conjunto electromecánico que comprende:
- Motorreductor trifásico, reductor de velocidad mediante mecanismo de tornillo sinfín-corona en acero estampado, con lubricación de por vida y con un alto rendimiento mecánico y bajo índice de ruidosidad.
- Variador de frecuencia (RF50-1000), permite programar diferentes velocidades de maniobra, incluidas la aceleración y el frenado.
- Transmisión secundaria mediante biela/manivela lo que garantiza una aceleración progresiva y frenado amortiguado del mástil. Este diseño asegura el bloqueo mecánico en las posiciones extremas.
- Interruptores de fin de carrera accionados por levas regulables angularmente.
- Protecciones eléctricas mediante disyuntor magnetotérmico regulable.
- Mástil de tubo de aluminio esmaltado blanco con bandas reflectantes a ambos lados.
- Bastidor en chapa de acero de 2mm de espesor.
- Puertas lateral y superior de acceso a los mecanismos para facilitar el mantenimiento. Bloqueadas mediante pasador interior y por cerraduras con 2 llaves.

Tratamiento de las superficies

El tratamiento anticorrosión del conjunto ha sido diseñado para resistir las condiciones atmosféricas más adversas.

Piezas Mecánicas:

- Tratadas por electrocincado bicromatado amarillo de espesor 10um.

Bastidor:

- Fosfatado de zinc de 5 micras.
- Tratamiento por cataforesis, procedimiento catódico, de espesor 20um.
- Capa de pintura de poliéster, de espesor 40um.
- Color estándar: RAL 1021(amarillo).

Opciones.

- Palanca de apertura manual en caso de fallo de corriente.
- Dispositivo electrónico de apertura de la barrera en caso de fallo de corriente.
- Fijación del mástil mediante tornillos de nylon para casos de impacto.
- Pintura no estándar (color RAL a definir).

Características Técnicas:

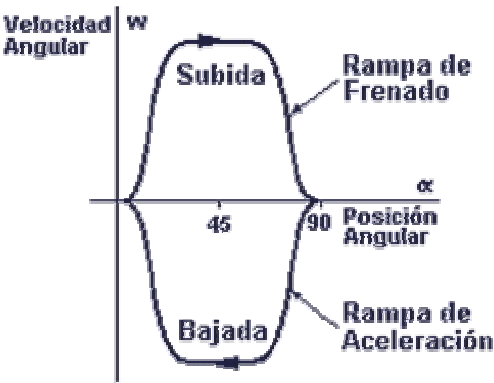
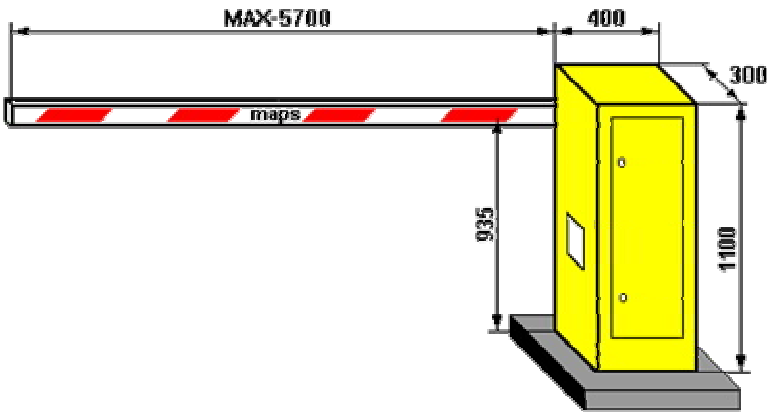
	CT70-850	RF50-1000
Tensión de Alimentación:	Trifásica 380VAC/220VAC,	Monofásica 220VAC,
.	50 (60)Hz	50(60)Hz
Motor eléctrico:	Síncrono Trifásico	Síncrono Trifásico
Potencia:	0,25CV	0,25CV
Velocidad Rotación:	850rpm	1000rpm
Par disponible:	ND.	ND.
Control interno:	Relés	Microprocesador
Aceleración/Desaceleración:	Por levas.	Definibles por intervalos tiempo
Protección Magnetotérmica:	Si, regulable.	Si, regulable.
Temperatura ambiente:	-20°C, 75°C	-20°C, 70°C
Peso:	45Kg	45Kg
Velocidad Apertura / Cierre:	1,5 segundos	0,6 segundos
Longitud Mástil:	5,7metros	5,7metros
Tratamiento Anticorrosión Bastidor:	Cataforesis	Cataforesis
Tratamiento Anticorrosión Mecanismo:	Baño electrolítico (zincado)	Baño electrolítico (zincado)
Pintura:	Poliéster RAL 1021	Poliéster RAL 1021
Grado de Protección:	IP-44	IP-44
entilación:	Rejilla (por convección)	Rejilla (por convección)
Mantenimiento:	Muy bajo	Nulo
Paro de emergencia:	Por Fotocélula	Por Fotocélula
Control Apertura / Cierre:	Por contacto libre potencial	Por contacto libre potencial
Resistencia mecánica (MCBF)		
(con mantenimiento normal):	5.000.000 ciclos	5.000.000 ciclos

Condiciones Comerciales:

Pedido mínimo:		
Plazo medio de entrega:		
Precio aprox. FF (IVA no incluido)		
con pértiga 3 metros:		
	2 unidades	2 unidades

4 semanas
A consultar

4 semanas
A consultar



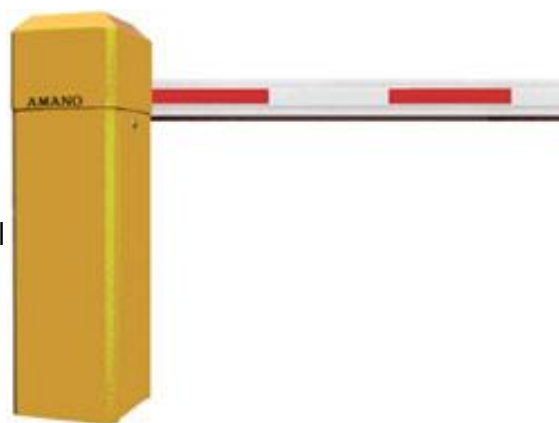
1.13. Barrera Automática TF-1203

CARACTERÍSTICAS TÉCNICAS:

- Alta velocidad de maniobra con un tiempo de 1.4 seg para Apertura/Cierre
- Diseño compacto y exclusivo
- Chasis preparado para la intemperie protegido con pintura estructural
- Control del motor con microprocesador
- Modos de operación seleccionables mediante interruptores
- Brazo de aluminio de 3 metros de longitud y sección rectangular
- Brazo intercambiable para funcionar en el lado derecho o izquierdo
- Movimiento del brazo suave y seguro de 90°
- Para uso intensivo de hasta 10.000 maniobras en 24 horas
- Interruptor de seguridad en la puerta con posición de mantenimiento
- Monitorización constante del ciclo de movimiento para la protección del motor
- Control electrónico de movimiento para inversión del mismo o paro en caso de colisión
- Cumple la normativa CE

Opciones

- Brazo articulado (máx.2.2m.)
- Brazo de 4m. de 6m. ó de 8m. de longitud
- Brazo tubular
- Receptor Radio Mando enchufable
- Apertura automática en caso de fallo de alimentación
- Apoyo electromagnético para la extremidad del brazo
- Luz de aviso cuando el brazo está en movimiento
- Anclaje exclusivo que facilita el montaje de la barrera



Características Eléctricas

- Suministro de energía : 230V AC 50Hz
- Consumo de energía: Máx. 1250W (sin calefactor)
- Conexión : carril Din Raíl con terminales atornillados
- Conexión: Manguera de cables numerados
- Entradas/Salidas a través de contacto seco
- Salida a 12 V DC

- Salida a 220/240 V AC

Características del motor

- Potencia del motor : 0.37 KW
- Amperios: 2.5 A
- Reductor del motor sin mantenimiento
- Elementos de transmisión con tratamiento anticorrosivo
- Probado para 10.000 maniobras diarias de Abertura/Cierre
- Tiempo de apertura/cierre de 1.4 segundos

Condiciones ambientales

- Temperatura de funcionamiento : de -20 ° hasta 70° C

Chasis

- Chasis de aluminio con impermeabilización IP54
- Medidas : 35cm (Ancho) x 30cm (Fondo) x110cm (Alto)
- Peso : 47,5 Kg
- Puerta frontal desmontable que facilita el acceso interno
- Pintura estructural de doble capa RAL1006 (Otros colores RAL están disponibles bajo pedido)

Brazo de barrera

- Brazo rectangular de aluminio (100mm x 25mm x 3000mm)
- Con bandas adhesivas reflectantes de color Rojo/Blanco
- Fácil intercambio del brazo del lado izquierdo al derecho
- Anclaje y fijación del brazo inequívocos
- Fácil compensación del peso del brazo mediante ajuste de la tensión de un muelle interno.

Lógica por microprocesador

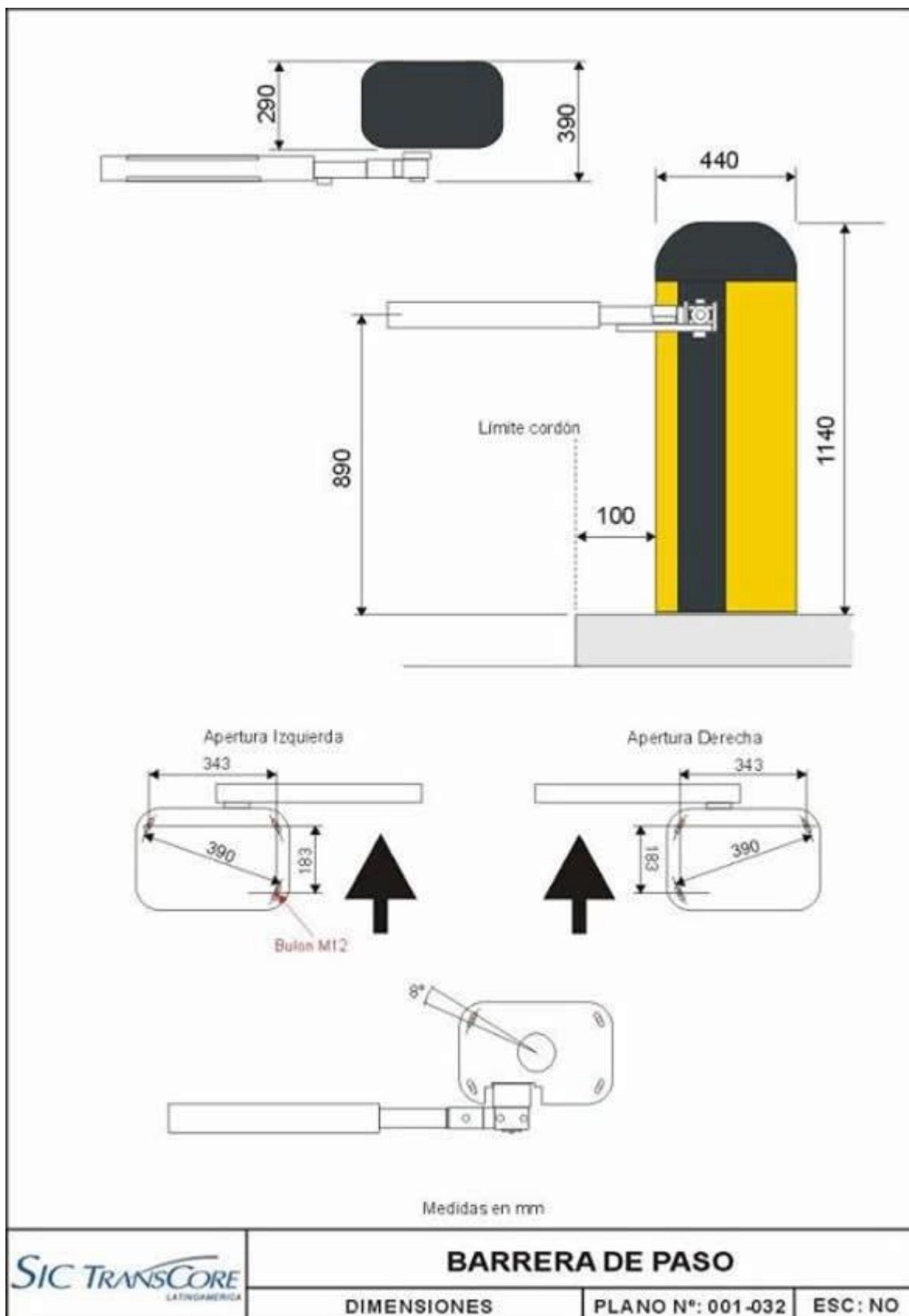
- Fácil conexionado y puesta en marcha
- Caja de Lógica extraíble que facilita el conexionado y la puesta en marcha
- Todas las funcionalidades tienen LED's de Estado y Control
- Modo de funcionamiento seleccionable con interruptores: Test, Manual, Semi y Totalmente automático
- Funcionalidades seleccionables con interruptores: Función del Lazo inductivo, Fotocélula de seguridad, Temporización de la maniobra de cierre, Inversión Automática del giro del motor

1.14. Sic-Transcore

CARACTERÍSTICAS TÉCNICAS

- Largo de brazo de: 1 a 12 metros
- Velocidad del ciclo:
1.56 seg (hasta 3 m), 9 seg (hasta 12 m)
 - Mantenimiento: solo revisión c/500.000 ciclos
 - Servicio: 8.000 ciclos/día
 - Con U.P.S. opera hasta 30 ciclos completos, [OPCIONAL]
- Brazo: circular de aluminio extruido
- Cobertura anti U V de P V C (termocontraíble)
- Acolchado de brazo: espuma de polietileno (blanco)
- Motor monofásico
- Tensión de alimentación: 2 x 220 V, consumo: 280 W
- Tensión de comando: 2 x 220 V , (opcionalmente otras tensiones)
- Señales secas para indicar la posición del brazo
- Gabinete en pintura poliéster para intemperie
- Zafe de brazo (para cortes de energía y embestidas)
- Brazo flotante (para evitar aplastamientos)
- Anclaje con registro para nivelación
- Regulación de posición del brazo
- Caja interna para cubrir el circuito eléctrico (opcional)
- No posee ningún elemento electrónico, salvo ups (opcional)
- Rango de temperatura de funcionamiento: de -5°C a 60°C, opcionalmente se puede aumentar de -30°C a 60°C
- Viento máx admisible en cualquier dirección: 50 km/hr, opcionalmente se puede elevar a 150 km/hr





1.15. Sistemas de visión artificial

Funcionamiento del Insignia 4

Sistema de Reconocimiento de Matrícula Portable Zamir dispone de un Sistema de Reconocimiento de Matrículas transportable en vehículo para aplicaciones en carretera.

El sistema Insignia MBL puede ser instalado en un vehículo.

Simplemente situando el vehículo en el lugar requerido, y colocando la unidad de cámara en

la vía de circulación que se pretende supervisar, el sistema Insignia MBL hará el resto. La unidad

se compone de cámara integrada, una unidad de iluminación día/noche, un PC industrial (opcional), monitor de pantalla plana y teclado.

Ver hoja de información específica de InSignia MBL.

Cámara a Color

Zamir ofrece una unidad de cámara opcional para la captura de imágenes en color o B/N junto con la imagen usada para el Reconocimiento de Matrícula. La imagen adicional está sincronizada en la base de datos con la imagen primaria de reconocimiento.

Ver la hoja de información específica de InSignia AVC.

Zonas de actuación

- Peajes
- Vigilancia
- Parking
- Seguridad interna
- Control acceso

Especificaciones Técnicas

Características del Sistema:

- 24 horas / día bajo cualquier condición ambiental.
- Responde ante un vehículo en menos de 0,2 s.
- Tasa de Reconocimiento:
- Nominal: > 93%
- Con apoyo de datos: > 98%

Condiciones de Instalación:

- Ancho de vía: Hasta 3,5 metros
- Distancia de controlador a Vehículo: 6 a 20 m.
- Angulo de Elevación: < 20°
- Angulo Lateral: 3° a 40°
- Angulo Inclinación: < 10°
- Velocidad vehículo: hasta 130 km. / hr

Controlador de Vía:

- Dimensiones: Largo x Ancho x Alto: 30 x 32 x 90 cm.
- Salidas Relé
- UIC Propietario con 4 N/A N/C
- Alimentación: 100 –110 V ó 220 –240 V
- Sensor Imagen: CCD B/N Alta resolución (CCIR System)
- Iluminación: Propietaria LED Unidad de Iluminación inteligente (IIU)

Condiciones Ambientales:

- Temperatura:
 - Operación: -10° C a 50° C
 - Almacén: -10° C a 60° C
- Humedad Relativa: 90% sin condensación.

Especificaciones Unidad de Proceso:

- Procesador: Pentium Intel Celeron 2 GHz
- RAM: 256 MB.
- Disco Duro: 40 GB.
- Sistema Operativo: MS Windows XP Pro
- Plataforma: InSignia 4.x Communication

Comunicaciones:

- 10 BaseT Ethernet TCP/IP
- Serie: RS232 o RS485 (opcional)
- USB (2 puertos)
- Radio: (opcional): GSM, SMS, Bluetooth, Firewire



1.16. Cámara fotográfica para ANPR

Disponer de imágenes del nivel apropiado de calidad es una cuestión crítica para un sistema de ANPR. Estas cámaras fotográficas de ultra baja iluminación B/N proporcionan alta resolución y alta intensidad incluso en las gamas IR (infra-rojas) para imágenes tanto durante el día como durante la noche. Con la cámara fotográfica para ANPR se incluye un dispositivo iluminador IR para proporcionar las condiciones óptimas de iluminación para tomar una imagen. El iluminador IR está construido con LED's de muy bajo consumo, y trabaja como un flash. Por ello, el consumo de energía es de solamente 20 W mientras que la potencia de la emisión luminosa es de 2000 W. Cuando el iluminador lanza su flash infra-rojo (IR), la cámara fotográfica toma una imagen de la matrícula del vehículo que pasa, que será transmitida como una señal de video analógica. Como el iluminador IR funciona en una longitud de onda imperceptible para el ojo humano no hay disturbio para los conductores. La cámara constituye una sola unidad incluyendo una cámara fotográfica, un flash IR de 2000 W, un filtro IR y un sincronizador especial diseñado y optimizado para las aplicaciones de la identificación de la placa del vehículo. El alcance efectivo del equipo es de 4 m hasta 13 m, pero puede variar, dependiendo de las condiciones de ambiente (tiempo, luz del sol, niebla, etc.) y de las características de las matrículas.

La cámara incluye los componentes siguientes:

- Cámara de alta resolución, ultra baja iluminación, B/N, para aplicación ANPR.
- Iluminador infrarrojo sincronizado con la cámara fotográfica de 2000 W eficaces.
- Lente de la alta calidad
- Cubierta intemperie
- Fuente de alimentación
- Caja de conexión externos, opcionales.

La cámara se ha diseñado especialmente para ANPR. Sin embargo se puede aplicar perfectamente en muchos otros usos donde es necesaria una calidad excepcional de la imagen.

**Especificaciones Técnicas:**

Peso	4.0 Kg.
Dimensiones (con la cubierta)	245 x 175 x 195 mm
Protección	IP 65
Alimentación eléctrica	20-40V CA C.C. o 15-28V (50-60Hz)
Consumo	15W
Longitud de onda infra-roja	850 nanómetro
Energía del pulso	2000W
Tiempo del obturador	1:100 000 s para los campos uniformes

1:10000 s para los campos irregulares

Señal de salida

PAL (NTSC opcional)

Tipo de sincronismo

Sincronismo interno

Cable

RG 59 coaxial

Fuente de alimentación

Opcional

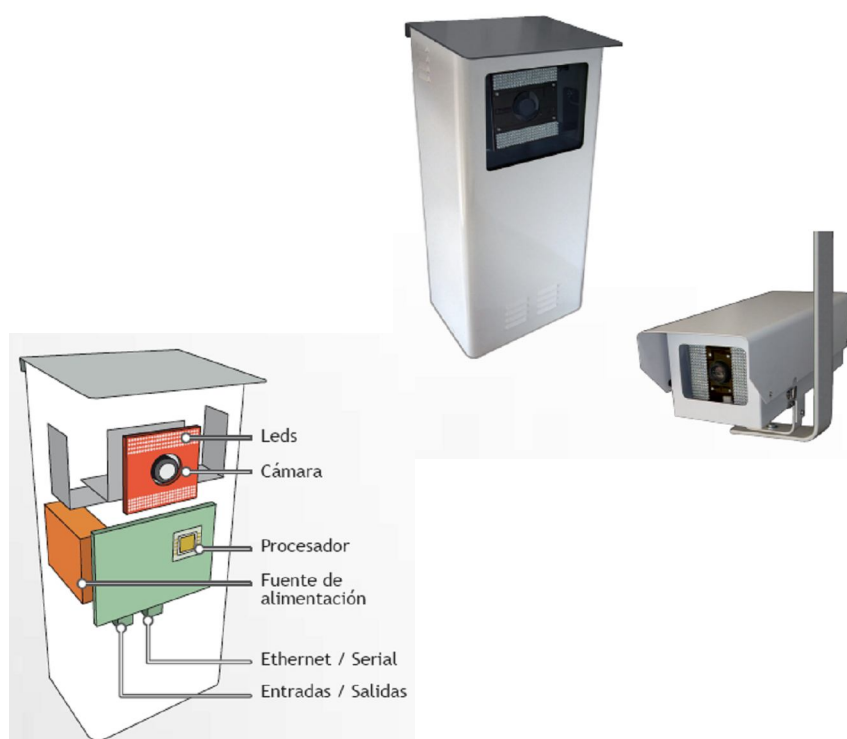
Caja de conexión externa

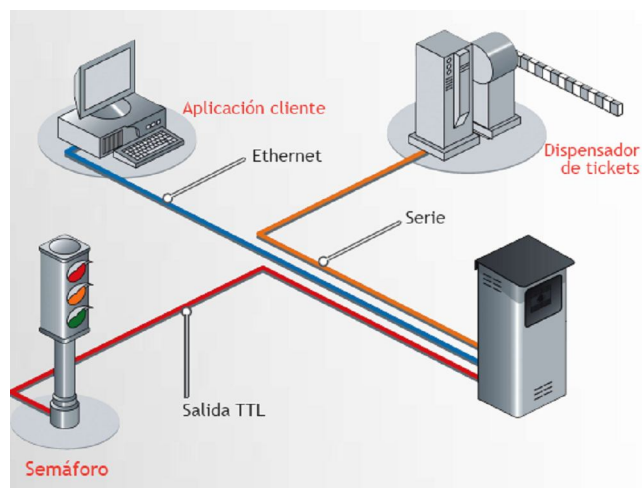
Opcional

1.17. **SmartLPR Access**

SmartLPR ACCESS es un equipo de lectura de matrículas diseñado para poder realizar el control de tráfico de vehículos en accesos con barrera.

Se trata de un dispositivo de lectura de matrículas All-in-one. Integra totalmente en un mismo equipo la iluminación, la cámara, el procesador, las entradas y salidas así como la fuente de alimentación. SmartLPR ACCESS se comunica mediante red Ethernet o comunicación serie.





SmartLPR ACCESS se comunica mediante red Ethernet o comunicación serie (RS-232/RS-485), dispone de 4 entradas y 4 salidas digitales que le permiten conectar y controlar diversos elementos, como sensores de presencia, semáforos, alarmas, etc.



ESPECIFICACIONES TÉCNICAS SmartLPR® ACCESS A

SmartLPR® ACCESS A SmartLPR® ACCESS WL A	
	Ref. 07000052 Ref. 07000053
Prestaciones	
Tiempo de respuesta:	De 0.10 a 0.75 segundos por vehículo. Tiempo de respuesta configurable según el número de imágenes capturadas y procesadas por vehículo.
Fiabilidad:	Hasta el 98%* (Ver High Reliability ^{QF} en apartado Características)
Matrículas reconocidas:	Europa: A, AND, B, BG, BY, CH, CZ, D, DK, E, EST, F, FIN, GB, GR, H, HR, I, IRL, L, LT, LV, M, MC, MK, N, NL, P, PL, RO, Rusia, S, SK, SLO, TR, UA. América: Argentina, Chile, Bolivia, Brasil, Colombia, Costa Rica, Cuba, Ecuador, El Salvador, Guatemala, Honduras, México, Nicaragua, Panama, Paraguay, Perú y Venezuela
Matrículas especiales:	Cuerpos diplomáticos y policiales, ejércitos y ministerios españoles
Protección :	IP53, equipo antivandálico
Condiciones de trabajo	
Anchura máxima de carril:	3.5 metros 5.5 metros
Ángulo de captura:	60° de ángulo horizontal y vertical respecto al plano de la matrícula
Operatividad:	24 horas al día, 365 días al año
Distancia máxima entre el equipo y la aplicación cliente:	La distancia máxima entre el equipo y la aplicación de gestión es la máxima que permite la red de comunicaciones
Condiciones Ambientales	
Temperatura de trabajo:	De -10° a 45° C
	--
Dimensiones	
Longitud:	26.0 cm
Anchura:	34.0 cm
Altura:	74.0 cm
Altura del soporte:	--
Iluminación	
Leds infrarrojos:	Sí
Vida útil:	Superior a 5 años
Cámara	
Scan progresivo:	Sí
Tamaño de imagen:	752 x 480 pixels 1024 x 480 pixels
Blanco y negro:	Sí
Consumo	
Consumo:	150 vatios aproximadamente
Conexiones	
Alimentación:	220 voltios
Puerto Ethernet:	10/100 Mbps
Puertos:	RS-232
Entradas/Salidas TTL:	4 entradas y 4 salidas digitales

ESPECIFICACIONES TÉCNICAS SmartLPR® ACCESS C

	SmartLPR® ACCESS C	SmartLPR® ACCESS WL C
	Ref. 07000050	Ref. 07000051
Prestaciones		
Tiempo de respuesta:	De 0.10 a 0.75 segundos por vehículo. Tiempo de respuesta configurable según el número de imágenes capturadas y procesadas por vehículo.	
Fiabilidad:	Hasta el 98%* (Ver High Reliability ^{QF} en apartado Características)	
Matrículas reconocidas:	Europa: A, AND, B, BG, BY, CH, CZ, D, DK, E, EST, F, FIN, GB, GR, H, HR, I, IRL, L, LT, LV, M, MC, MK, N, NL, P, PL, RO, Rusia, S, SK, SLO, TR, UA. América: Argentina, Chile, Bolivia, Brasil, Colombia, Costa Rica, Cuba, Ecuador, El Salvador, Guatemala, Honduras, México, Nicaragua, Panama, Paraguay, Perú y Venezuela	
Matrículas especiales:	Cuerpos diplomáticos y policiales; ejércitos y ministerios españoles; materias peligrosas	
Protección:	IP65	
Condiciones de trabajo		
Anchura máxima de carril:	3.5 metros	5.5 metros
Ángulo de captura:	60° de ángulo horizontal y vertical respecto al plano de la matrícula	
Operatividad:	24 horas al día, 365 días al año	
Distancia máxima entre el equipo y la aplicación cliente:	La distancia máxima entre el equipo y la aplicación de gestión es la máxima que permite la red de comunicaciones	
Condiciones Ambientales		
Temperatura de trabajo:	De -10° a 45° C	
Dimensiones		
Longitud:	59.5 cm	
Anchura:	22.5 cm	
Altura:	19.0 cm	
Altura del soporte:	62.5 cm	
Iluminación		
Leds infrarrojos:	Sí	
Vida útil:	Superior a 5 años	
Cámara		
Scan progresivo:	Sí	
Tamaño de imagen:	640 x 480 pixels	1024 x 480 pixels
Blanco y negro:	Sí	
Consumo		
Consumo:	100 vatios aproximadamente	
Conexiones		
Alimentación:	220 voltios	
Puerto Ethernet:	10/100 Mps	
Puertos:	RS-232/RS-485	
Entradas/Salidas TTL:	4 entradas y 4 salidas digitales	

1.18. Mueble de control para lectura de matrículas PNA- C4

Características

Chasis metálico resistente para intemperie

Fácil montaje utilizando un anclaje metálico

Doble lazo inductivo integrado

Electrónica de control programable remotamente

Interface bus CAN 2.0 integrada

Elementos reconocimiento de matrículas:

- Cámara CCD digital color
- Iluminador halógeno 50W 10°

Integración con el sistema de parking existente (expendedora o validadora)

Interface entrada/salida para el control de la barrera

PNA-C4

Mueble de Control para Lectura de Matrículas



Opciones

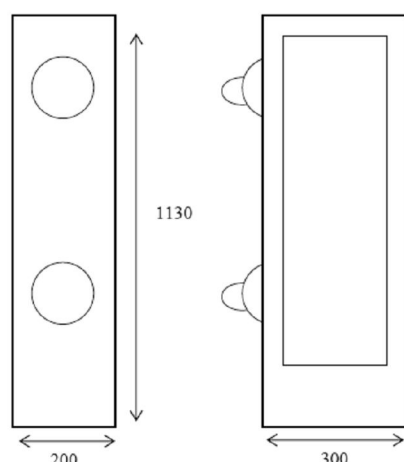
- Color personalizado bajo pedido (por defecto RAL 1004)

Los muebles de control *masterASP PNA-C4* proporcionan las funciones necesarias en un sistema de lectura de matrículas. Constan de una cámara digital de color adecuada para leer matrículas y un iluminador que se activa solo en caso necesario.

Disponen de un puerto de comunicaciones para la integración con las máquinas de parking existentes. Además, a través de su interface de entrada/salida permiten controlar una barrera o elemento similar.

Gracias a la interface de comunicaciones CAN 2.0, se obtiene una comunicación robusta con el sistema central y la posibilidad de actualizar y configurar el firmware.

Medidas



Especificaciones Eléctricas

Alimentación: monofásica 220V/50Hz
Consumo: 65 W (con halógena de 50W)

Conexión:

- Interface CAN con central: RJ45
- Video: RG59
- Bus de control para expendedora/validadora
- Bus de control para barrera

Electrónica y Configuración

La placa de control contiene interruptores de test y configuración así como LEDs de estado. Conectando un LCD o a través del bus CAN se puede modificar su funcionamiento.

Cámara:

- Sensor CCD infrarojos de alta sensibilidad
- Imagen B/N en poca iluminación automático
- Zoom 23x óptico, 10x digital
- Controlable remotamente

Electrónica:

- Potencia sobre el iluminador (PWM)
- Velocidad del bus CAN
- Firmware actualizable remotamente

Temperatura de funcionamiento de -20°C hasta 70°C.

Chasis

Chasis fabricado en plancha de hierro galvanizado y posterior pintado al horno con doble capa en color amarillo RAL1004.

Dim: 30cm Ancho, 20cm Fondo, 113cm Alto
Peso: 30Kg
Puerta delantera con cerradura de seguridad

1.19. Tarjetas para captura de vídeo

TARJETAS FXVD4 PARA CAPTURA DE VIDEO

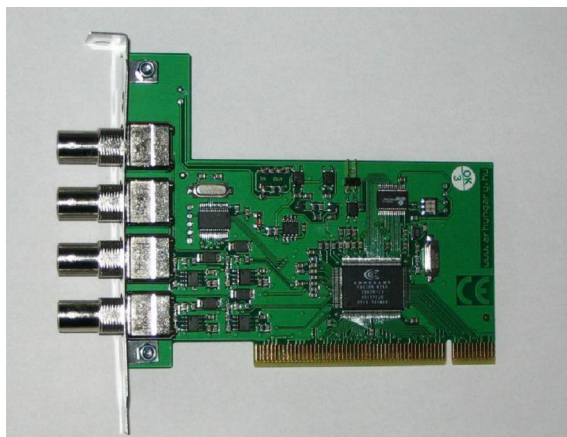
Tarjeta de video en formato PCI y Tarjeta de video en formato PC104+ son unas tarjetas de alta calidad de 4 canales para captura de video, desarrollada especialmente para el sistema "ANPR" (reconocimiento automático del número de placa). Su estabilidad y alto rendimiento la hacen ideal también para otros varios usos. Se basan en el "chip" de captura de video Bt878 cuya arquitectura avanzada proporciona imágenes de la alta calidad. Los servicios de este "chip" son soportados por módulos de hardware adicionales situados en la tarjeta, permitiendo que se alcance un rendimiento excepcional. Aparte de su función primaria de convertir la señal a video digital, el "chip" se comunica con el coprocesador RISC de la tarjeta. Este coprocesador funciona como un hardware integrado para la protección del software "ANPR". Una función importante de esta protección es parte importante de la tarjeta de video. Con su ayuda la tarjeta puede reiniciar el ordenador en caso de fallos del sistema, lo cual mejora considerablemente la estabilidad y la confiabilidad de sistemas aislados. La tarjeta ha sido desarrollada para manejar señales analógicas de video tanto en PAL como en NTSC, recibidas a través de alguno de los cuatro conectores BNC de entrada. Los tiempos de conmutación entre los canales de entrada de la tarjeta son del orden de 30 milisegundos, lo cual permite al usuario construir sistemas secuenciales de alta velocidad que manejan las señales de cuatro canales video casi simultáneamente.

Están disponibles tres tipos de tarjetas de video:

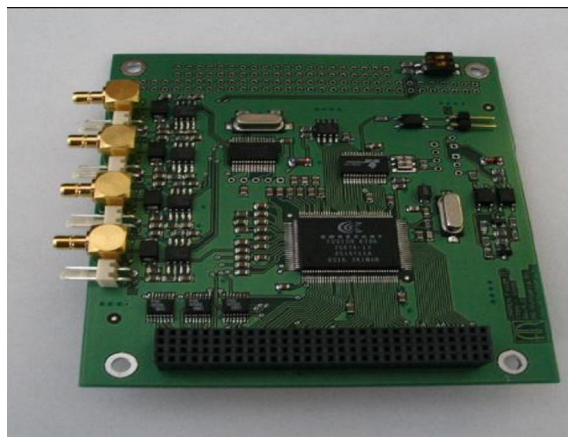
- Tarjeta (Flujo Libre): disponibles funciones para captura de video y cooperación con el software "Flujo Libre", en formatos PCI o PC104+.
- Tarjeta (Parking): disponibles funciones para captura de video y cooperación con el software "Parking", en formatos PCI o PC104+.
- Tarjeta (Especial): disponibles funciones para captura de video. Recomendada para ser utilizada como tarjeta adicional con alguna de las anteriores si se requieren más de 4 entradas de video. En formatos PCI o PC104+.

La tarjeta FXVD4 es también una solución ideal para:

- Captación de imagen industrial y de maquinaria.
- Seguridad y control industrial.
- Video-vigilancia digital.
- Producción en cadena.
- Robótica.
- Microscopía y medicina.
- Internet y aplicaciones multimedia.



Formato PCI



Formato PC104+

ESPECIFICACIONES TÉCNICAS

Especificaciones técnicas Tarjeta de video formato PCI:

Interface	PCI 2.1
Conversión A/D	Bt 878
Entrada	4 x BNC vídeo, 75 ohm
Protección	Programable con 1-250 sec. de retardo seleccionable
Sistema operativo soportado	Windows NT 4.0, Windows 2000/2003, Windows XP
Formatos de video	PAL o NTSC (75 W, 1 Vp-p)
Tramas de video	Full", "Field" "Half @ any / odd / even"

Salida de imagen	Memoria, archivo
Tipos de archivos	BMP, JPEG
Formatos de archivos	Grayscale, RGB15, RGB16, RGB24, RGB32, YUV422
Documentación	Manual de referencia en formato electrónico
Requisitos del sistema	Slot PCI libre
Consumo de energía	Aprox. 450 mA
Tamaño	97 x 120 mm

Especificaciones técnicas Tarjeta de video formato PCI104+:

Interface	PC104+
Conversión A/D	Bt 878
Entrada	4 x BNC video, 75 ohm
Protección	Programable con 1-250 sec. de retardo seleccionable
Sistema operativo soportado	Windows NT 4.0, Windows 2000/2003, Windows XP
Formatos de video	PAL o NTSC (75 W, 1 Vp-p)
Tramas de video	Full", "Field" "Half @ any / odd / even"
Salida de imagen	Memoria, archivo

Tipos de archivos	BMP, JPEG
Formatos de archivos	Grayscale, RGB15, RGB16, RGB24, RGB32, YUV422
Consumo de energía	Aprox. 450 mA
Tamaño	98.17 x 95.885 mm

ANEXO 2

PROGRAMACIÓN EN CY C++ DE LA APLICACIÓN

2.1. Programación en C.

```
#include <mil.h>

#include <conio.h>

#include <stdio.h>

#include <stdlib.h>

#include <string.h>


/* Especificaciones de librerias MIL */


#define IMAGE_FILE_DEFINITION M_IMAGE_PATH MIL_TEXT("Patron.bmp")

/* caracteres de la imagen fuente para la calibracion de la imagen */

#define TEXT_DEFINITION "012345678 9ABCDEFGHI JKLMNOPQR STUVWXYZ"

/* Font normalization size Y. */


/* Máxima longitud de matricula */

#define STRING_MAX_SIZE 32L

/* Main */

int main()

{

    MIL_ID MilApplication, /* Identificador de aplicacion */

    MilSystem, /* Identificador de sistema */

    MilDisplay, /* Identificador de display */

    Millmage, /* Identificador del buffer de imagen */

    MilOverlayImage, /*Identificador de imagen superpuesta */
```

```
MilStrContext, /*Identificador de cadena de contexto */
MilStrResult, /* Identificador del buffer de la cadena resultado */
MillImageDig,
MilDigitizer;

long NumberOfStringRead; /* Numero de cadenas leidas */
double Score; /* % de correspondencias*/
char StringResult[STRING_MAX_SIZE+1]; /*Cadena de caracteres leida */
char cadena[100];
double Time = 0.0; /* Tiempo */
int x=0;
long int final,comienzo; /* Variables de acceso para manejar ficheros*/
FILE *fich;
/*Prueba de escritura por pantalla */
printf("\nLectura de matriculas:\n");
printf("-----\n\n");

/* Localizaciones de memoria por defecto */
//for(i=8;i<50;i++)
//{
    MappAllocDefault(M_SETUP, &MilApplication, &MilSystem,
                    &MilDisplay, &MilDigitizer, &MillImageDig);
    /* Grabacion continua de entrada al parking */
    MdigGrabContinuous(MilDigitizer, MillImageDig);
    /* Al pulsar se congelará la imagen */
```

```
printf("\nAdquisicion de imagen preparanda:\n");

printf("-----\n\n");

printf("Grabacion continua en proceso.\n");

printf("Pulsa para parar\n\n");

getch();

/* grabacion parada. */

MdigHalt(MilDigitizer);

/* Mostrando resultado */

printf("Grabacion continua parada\n\n");

printf("Pulsa para adquirir imagen\n\n");

getch();

/* La imagen adquirida se envia a buffer */

MdigGrab(MilDigitizer, MillImageDig);

/* Mostrando resultados */

printf("La imagen obtenida es:\n");

printf("Pulsa para finalizar\n\n");

getch();

/* Se carga la imagen patron */

MbufRestore(IMAGE_FILE_DEFINITION, MilSystem, &MillImage);

/* Se dispone a visualizar la imagen en un único display */

MdispSelect(MilDisplay, MillImage);

MdispControl(MilDisplay, M_OVERLAY, M_ENABLE);

MdispInquire(MilDisplay, M_OVERLAY_ID, &MilOverlayImage);

/* Se reserva memoria para una nueva cadena control */
```

```
MstrAlloc( MilSystem, M_FEATURE_BASED, M_DEFAULT, &MilStrContext);
```

```
/* Se reserva memoria para la cadena de control resultado */
```

```
MstrAllocResult(MilSystem, M_DEFAULT, &MilStrResult);
```

```
MstrControl(MilStrContext, M_CONTEXT, M_FONT_ADD, M_USER_DEFINED);
```

```
/* Se comprueba la imagen patron con la cadena introducida */
```

```
MstrEditFont(MilStrContext, M_FONT_INDEX(0), M_CHAR_ADD,
```

```
M_USER_DEFINED + M_FOREGROUND_BLACK,
```

```
MillImage, TEXT_DEFINITION, M_NULL);
```

```
/* Se pintan los caracteres encontrados en la imagen patron */
```

```
MgraColor(M_DEFAULT, M_COLOR_GREEN);
```

```
MstrDraw(M_DEFAULT, MilStrContext, MilOverlayImage, M_DRAW_CHAR,
```

```
M_FONT_INDEX(0), M_NULL, M_ORIGINAL);
```

```
/* Se añade 1 nueva cadena vacia modelo de la matricula */
```

```
MstrControl(MilStrContext, M_CONTEXT, M_STRING_ADD, M_USER_DEFINED);
```

```
/*Numero de cadenas a leer */
```

```
MstrControl(MilStrContext, M_CONTEXT, M_STRING_NUMBER, 1);
```

```
/* Numero de caracteres max y min de la cadena a leer */
```

```
MstrControl(MilStrContext, M_STRING_INDEX(M_ALL), M_STRING_SIZE_MIN, 7);
```

```
MstrControl(MilStrContext, M_STRING_INDEX(M_ALL), M_STRING_SIZE_MAX, 8);
```

```
/* Solo letras mayusculas */
```

```
MstrSetConstraint(MilStrContext, M_STRING_INDEX(0), M_DEFAULT, M_LETTER +  
M_UPPERCASE, M_NULL);
```

```
/* Se fijan el tipo de caracteres de la cadena modelo */
```

```
MstrSetConstraint(MilStrContext, M_STRING_INDEX(0), 0, M_ANY, M_NULL);
```

```
MstrSetConstraint(MilStrContext, M_STRING_INDEX(0), 1, M_ANY, M_NULL);
```

```
MstrSetConstraint(MilStrContext, M_STRING_INDEX(0), 2, M_ANY, M_NULL);
```

```
MstrSetConstraint(MilStrContext, M_STRING_INDEX(0), 3, M_ANY, M_NULL);
```

```
MstrSetConstraint(MilStrContext, M_STRING_INDEX(0), 4, M_ANY, M_NULL);
```

```
MstrSetConstraint(MilStrContext, M_STRING_INDEX(0), 5, M_ANY, M_NULL);
```

```
MstrSetConstraint(MilStrContext, M_STRING_INDEX(0), 6, M_ANY, M_NULL);
```

```
MstrSetConstraint(MilStrContext, M_STRING_INDEX(0), 7, M_ANY, M_NULL);
```

```
printf("Este programa ha tomado como imagen fuente matriculas españolas.\n");
```

```
printf("Pulsa para continuar\n\n");
```

```
getch();
```

```
/* Se elimina la imagen cargada en el display */
```

```
MdispControl(MilDisplay, M_OVERLAY_CLEAR, M_DEFAULT);
```

```
/* se carga la captura de la cámara cargando sólo una capa para obtener una imagen en  
escala de grises */
```

```
MbufCopyColor(MilImageDig,MilImage,M_RED);

MstrPreprocess(MilStrContext, M_DEFAULT);

MstrRead(MilStrContext, MilImage, MilStrResult);


/* Se resetea el reloj */


MappTimer(M_TIMER_RESET+M_SYNCHRONOUS, M_NULL);


MstrRead(MilStrContext, MilImage, MilStrResult);


/* se lee el reloj para obtener la duracion del proceso */


MappTimer(M_TIMER_READ+M_SYNCHRONOUS, &Time);


/*Obtiene el numero de cadenas leidas y lo muestra*/


MstrGetResult(MilStrResult, M_GENERAL, M_STRING_NUMBER + M_TYPE_LONG,
&NumberOfStringRead);

if( NumberOfStringRead >= 1)
{
printf("Matricula leida en: (%.2lf ms)\n\n", Time*1000 );

/* Draw read result. */

MgraColor(M_DEFAULT, M_COLOR_BLUE);

MstrDraw(M_DEFAULT, MilStrResult, MilOverlayImage, M_DRAW_STRING, M_ALL,
M_NULL, M_DEFAULT);
```

```
MgraColor(M_DEFAULT, M_COLOR_GREEN);

MstrDraw(M_DEFAULT, MilStrResult, MilOverlayImage, M_DRAW_STRING_BOX,
M_ALL, M_NULL, M_DEFAULT);

/* Saca por pantalla la matricula leída. */

printf(" Matricula: \n" );

printf(" ----- \n" );

MstrGetResult(MilStrResult, 0, M_STRING, StringResult);

MstrGetResult(MilStrResult, 0, M_STRING_SCORE, &Score);

printf(" %s %.1f\n", StringResult, Score );

}

else

{

printf("Error: Matricula no leida.\n");

}

char reg;

char leer;

FILE *f = fopen("a.txt", "r+b");

while (fgets(cadena, 100, f) != NULL)

{

if (strcmp(cadena, StringResult) == 0)

{

printf("Son iguales");

printf(" %s %.1f\n",cadena);

x++;

}
```

```
printf(" La variable X = ",x);
```

```
//fseek(f, 0L, SEEK_SET);
```

```
//final = ftell( f );
```

```
//fwrite(&reg ,0L, 1, f);
```

```
//printf(" %s %.1f\n",final);
```

```
}
```

```
if( (comienzo=ftell( f )) < 0 ) printf( "ERROR: ftell no ha funcionado\n" );
```

```
else printf( "Posicion del fichero: %d\n\n", comienzo );
```

```
fseek( f, 0L, SEEK_END );
```

```
final = ftell( f );
```

```
printf( "Posicion del final fichero: %d\n\n", final );
```

```
fread( &leer, comienzo, final, f );
```

```
printf( "%d\t", &leer );
```

```
}
```

```
printf("\nPulsar para salir.\n\n");
```

```
getch();
```

```
/* Libera espacios de memoria */
```

```
MstrFree(MilStrContext);

MstrFree(MilStrResult);

MbufFree(MilImage);


MappFreeDefault(MilApplication, MilSystem, MilDisplay, M_NULL, M_NULL);

}

}
```

2.2. Programación en C++

```
// MStringReaderDlg.cpp : implementation file
//
```

```
#include "stdafx.h"
#include "MStringReader.h"
#include "MStringReaderDlg.h"
```

```
#ifdef _DEBUG
#define new DEBUG_NEW
#endif
```

```
////////////////////////////////
```

```
// ActiveMIL constants

//

// Location of images files

//

const char IMAGE_FILE_DEFINITION[] = "patron.bmp";

const char IMAGE_FILE_TO_READ[] = "matricula4.bmp";


// String containing all characters used for font definition.

//

const char TEXT_DEFINITION[] = "012345678 9ABCDEFGHI JKLMNOPQR STUVWXYZ";


// Font normalisation size Y

//

//const int NORMALIZATION_SIZE_Y = 20;


// Max size of plate string

//

const int STRING_MAX_SIZE = 32;


// Standard display width

//

const int DEFAULT_DISPLAY_WIDTH = 480;


// CAboutDlg dialog used for App About
```

```
class CAboutDlg : public CDialog
{
public:
    CAboutDlg();

// Dialog Data
    enum { IDD = IDD_ABOUTBOX };

protected:
    virtual void DoDataExchange(CDataExchange* pDX); // DDX/DDV support

// Implementation
protected:
    DECLARE_MESSAGE_MAP()
};

CAboutDlg::CAboutDlg() : CDialog(CAboutDlg::IDD)
{
}

void CAboutDlg::DoDataExchange(CDataExchange* pDX)
{
    CDialog::DoDataExchange(pDX);
}
```

```
BEGIN_MESSAGE_MAP(CAboutDlg, CDialog)
```

```
END_MESSAGE_MAP()
```

```
// CMStringReaderDlg dialog
```

```
CMStringReaderDlg::CMStringReaderDlg(CWnd* pParent /*=NULL*/)
{
    : CDialog(CMStringReaderDlg::IDD, pParent)
    , m_CommentText("")
    , m_StepNumber(1)
    {
        m_hIcon = AfxGetApp()->LoadIcon(IDR_MAINFRAME);
    }
}

void CMStringReaderDlg::DoDataExchange(CDataExchange* pDX)
{
    CDialog::DoDataExchange(pDX);

    DDX_Control(pDX, IDC_APPLICATION1, m_Application1);
    DDX_Control(pDX, IDC_SYSTEM1, m_System1);
    DDX_Control(pDX, IDC_IMAGE1, m_Image1);
    DDX_Control(pDX, IDC_DISPLAY1, m_Display1);
}
```



```
DDX_Control(pDX, IDC_GRAPHICCONTEXT1, m_GraphicContext1);  
DDX_Control(pDX, IDC_STRINGREADER1, m_StringReader);  
DDX_Text(pDX, IDC_COMMENTTEXT, m_CommentText);  
DDX_Control(pDX, IDC_DIGITIZER3, m_Digitizer);  
}
```

```
BEGIN_MESSAGE_MAP(CMStringReaderDlg, CDialog)  
    ON_WM_SYSCOMMAND()  
    ON_WM_PAINT()  
    ON_WM_QUERYDRAGICON()  
    /}}AFX_MSG_MAP  
    ON_BN_CLICKED(IDC_NEXTBUTTON, OnBnClickedNextbutton)  
END_MESSAGE_MAP()
```

```
// CMStringReaderDlg message handlers
```

```
BOOL CMStringReaderDlg::OnInitDialog()  
{  
    CDialog::OnInitDialog();  
  
    // Add "About..." menu item to system menu.  
  
    // IDM_ABOUTBOX must be in the system command range.  
    ASSERT((IDM_ABOUTBOX & 0xFFF0) == IDM_ABOUTBOX);
```

```
ASSERT(IDM_ABOUTBOX < 0xF000);
```

```
CMenu* pSysMenu = GetSystemMenu(FALSE);
if (pSysMenu != NULL)
{
    CString strAboutMenu;
    strAboutMenu.LoadString(IDS_ABOUTBOX);
    if (!strAboutMenu.IsEmpty())
    {
        pSysMenu->AppendMenu(MF_SEPARATOR);
        pSysMenu->AppendMenu(MF_STRING, IDM_ABOUTBOX, strAboutMenu);
    }
}
```

```
////////////////////////////////////
```

```
// ActiveMIL example initialization
```

```
    CString    strActiveMILImagesPath    =    static_cast<LPCTSTR>(m_Application1-
>ActiveMILImagesPath);
```

```
CString strActiveMILImageFile = strActiveMILImagesPath + IMAGE_FILE_DEFINITION;
```

```
m_Image1->Load(_bstr_t(strActiveMILImageFile), FALSE);
```

```
AdjustDialog(strActiveMILImageFile);
```

```
// Draw the characters of the font
```

```
//

    IStringReaderFontCharactersPtr CharactersPtr = m_StringReader->Fonts->Item(1)-
>Characters;

    CharactersPtr->AddFromImage(m_Image1, strForegroundBlack, TEXT_DEFINITION);

    CharactersPtr->Draw(m_GraphicContext1,
static_cast<StrDrawCharactersOperationConstants>(strDrawCharacters          +
strCharactersOriginalPosition));

    // Normalize the characters of the font to an appropriate size.

    //
    //CharactersPtr->Normalize(strNormalizeReferenceSizeY,      NORMALIZATION_SIZE_Y,
_bstr_t(""));

    //////////////////////////////////////

    // ActiveMIL example initialization

    //

    // Resize the dialog

    //

    // AdjustDialog();

    // Start grabbing continuously

    //

    m_Digitizer->GrabContinuous();
```

```
m_CommentText = "This program has defined a font with this image of Quebec plates  
mosaic.\n"
```

```
    "Click <Next> or press <Enter> to continue.";
```

```
    UpdateData(FALSE);
```

```
    return TRUE; // return TRUE unless you set the focus to a control
```

```
}
```

```
void CMStringReaderDlg::OnSysCommand(UINT nID, LPARAM lParam)
```

```
{
```

```
    if ((nID & 0xFFF0) == IDM_ABOUTBOX)
```

```
    {
```

```
        CAboutDlg dlgAbout;
```

```
        dlgAbout.DoModal();
```

```
    }
```

```
    else
```

```
    {
```

```
        CDialog::OnSysCommand(nID, lParam);
```

```
    }
```

```
}
```

```
// If you add a minimize button to your dialog, you will need the code below
```

```
// to draw the icon. For MFC applications using the document/view model,
```

```
// this is automatically done for you by the framework.
```

```
void CMStringReaderDlg::OnPaint()
{
    if (IsIconic())
    {
        CPaintDC dc(this); // device context for painting

        SendMessage(WM_ICONERASEBKGND, reinterpret_cast<WPARAM>(dc.GetSafeHdc()),
0);

        // Center icon in client rectangle
        int cxIcon = GetSystemMetrics(SM_CXICON);
        int cyIcon = GetSystemMetrics(SM_CYICON);
        CRect rect;
        GetClientRect(&rect);
        int x = (rect.Width() - cxIcon + 1) / 2;
        int y = (rect.Height() - cyIcon + 1) / 2;

        // Draw the icon
        dc.DrawIcon(x, y, m_hIcon);
    }
    else
    {
        CDialog::OnPaint();
    }
}
```

```
// The system calls this function to obtain the cursor to display while the user drags
```

```
// the minimized window.
```

```
HCURSOR CMStringReaderDlg::OnQueryDragIcon()
```

```
{
```

```
    return static_cast<HCURSOR>(m_hIcon);
```

```
}
```

```
////////////////////////////////////
```

```
// ActiveMIL procedure: In response of clicking the "Next" button.
```

```
//          Drives the example through each step,
```

```
//          performing appropriate actions
```

```
//
```

```
void CMStringReaderDlg::OnBnClickedNextbutton()
```

```
{
```

```
    // Increment the step counter and updates text in the dialog
```

```
    //
```

```
    AdjustStepNumber();
```

```
    switch(m_StepNumber)
```

```
    {
```

```
        case 2:
```

```
        {
```

```
            // Clear the display's overlay image
```

```
            //
```

```
m_Display1->ClearOverlay(dispClearOverlayDefault);

// Load the image to read
//

CString strActiveMILImagesPath = static_cast<LPCTSTR>(m_Application1-
>ActiveMILImagesPath);

CString strActiveMILImageFile = strActiveMILImagesPath + IMAGE_FILE_TO_READ;

    m_Image1->Clear(0);

//m_Image1->Copy(m_Image1,imAllBands);

    //m_Image1->Load(_bstr_t(strActiveMILImageFile), FALSE);

    m_Digitizer->Halt();

AdjustDialog(strActiveMILImageFile);

double ReadTime = 0.0;

// Preprocess the StringReader control
//

m_StringReader->Preprocess();

// Dummy first read for better function timing accuracy (model cache effet,...)
//

m_StringReader->Read();

// Reset the timer
```

```
//  
  
m_Application1->Timer->SynchronizationMode = appTimerSynchronous;  
m_Application1->Timer->Reset();  
  
// Perform the read operation in the specified target image.  
  
//  
m_StringReader->Read();  
  
// Read the timer  
  
//  
ReadTime = m_Application1->Timer->Read();  
  
IStringReaderResultsPtr Results = m_StringReader->Results;  
if(Results->NumberOfStrings >= 1)  
{  
    // Display the results  
  
    //  
    m_CommentText.Format("The license plate was read successfully (%.2lf ms)\n"  
        "String : %s      Score : %.1lf%%\n",  
        ReadTime * 1000,  
        static_cast<LPCTSTR>(Results->Strings->Item(1)->Value),  
        Results->Strings->Item(1)->Score);  
  
    // Draw the string box  
  
    //
```



```
Results->Draw(m_GraphicContext1, strDrawResultsStringsBoxes);

// Change the color of the graphic context and draw the string
//
m_GraphicContext1->ForegroundColor = RGB(0, 0, 255);
Results->Draw(m_GraphicContext1, strDrawResultsStrings);
}
else
{
    m_CommentText = "Error: Plate was not read.\n";
}

// Update comment text
//
m_CommentText += "Click <Next> or press <Enter> to end.";
UpdateData(FALSE);
break;
}

default :

    // Last Step: End the application
    //
    // Close dialog application
    //
    EndDialog(IDOK);
    break;
```

```
    }  
}  
  
////////////////////////////////////  
// ActiveMIL procedure: Increments the step counter and updates text in  
//           the dialog according to the new value  
//  
void CMStringReaderDlg::AdjustStepNumber()  
{  
    // Increment the step count  
    //  
    m_StepNumber++;  
  
    // Write the step count in the comment frame  
    //  
    CString str;  
    str.Format("Step %d:", m_StepNumber);  
    CWnd *pStepNumber = GetDlgItem(IDC_STEPNUMBER);  
    pStepNumber->SetWindowText(str);  
}  
  
////////////////////////////////////  
// ActiveMIL procedure :   Calculate how to resize the dialog in order to  
//           view all element in all resolutions and font sizes  
//
```

```
void CMStringReaderDlg::AdjustDialog(CString ImagePath)
{
    // Retrieve the relevant dialog controls

    //

    CWnd *pNextButton = GetDlgItem(IDC_NEXTBUTTON);
    CWnd *pStepNumber = GetDlgItem(IDC_COMMENTTEXT);


    int MimlImageSizeX = m_Image1->SizeX;
    int MimlImageSizeY = m_Image1->SizeY;
    long nMaxDisplayWidth;
    long nMaxNextStepWidth;
    CRect rectNextButton;
    CRect rectCommentFrame;
    CRect rectDisplay;
    CRect rectDialog;


    // Adjust the size of the Display
    //

    m_Display1.SetWindowPos(NULL,                // pWndInsertAfter
        0,                // x
        0,                // y
        MimlImageSizeX,    // cx
        MimlImageSizeY,    // cy
        SWP_NOMOVE | SWP_NOOWNERZORDER); // flags
```

```
// Get the sizes of relevant elements and calculate how to resize the
// dialog in order to view all elements, in all resolutions and font sizes
//
pNextButton->GetWindowRect(&rectNextButton);
pStepNumber->GetWindowRect(&rectCommentFrame);
m_Display1.GetWindowRect(&rectDisplay);
GetWindowRect(&rectDialog);

// Check the width of the display, if it is big enough, we will
// adjust the comment text frame and the next button
// to fit the width of the display
//
if(rectDisplay.Width() > DEFAULT_DISPLAY_WIDTH)
{
    // Calculate the size of the window displacements
    //
    int SizeChange = rectDisplay.right - rectNextButton.right;

    // Change the sizes of the Next button and the comment frame
    //
    rectNextButton.OffsetRect(SizeChange, 0);
    rectCommentFrame.InflateRect(0, 0, SizeChange, 0);

    // Move the windows
    //
```

```
ScreenToClient(&rectNextButton);
ScreenToClient(&rectCommentFrame);

pNextButton->MoveWindow(&rectNextButton, FALSE);
pStepNumber->MoveWindow(&rectCommentFrame, FALSE);

// Reconvert the rectangles to screen coordinates
//
ClientToScreen(&rectCommentFrame);
ClientToScreen(&rectNextButton);
}

// Now that the next button and comment frame are in their correct positions
// We can resize the dialog
//
nMaxDisplayWidth = rectDisplay.Width() +
    2 * abs(rectDialog.left - rectDisplay.left);

nMaxNextStepWidth = rectNextButton.left - rectDialog.left +
    rectNextButton.Width() +
    rectCommentFrame.left - rectDialog.left;

if (nMaxNextStepWidth > nMaxDisplayWidth)
{
    SetWindowPos(NULL,                                // pWndInsertAfter
```

```
    0,                // x
    0,                // y
    nMaxNextStepWidth,    // cx
    rectDisplay.Height() +
    rectDisplay.top - rectDialog.top +
    rectDisplay.left - rectDialog.left,    // cy
    SWP_SHOWWINDOW|SWP_NOMOVE);    // flags
}
else
{
    SetWindowPos(NULL,    // pWndInsertAfter
    0,                // x
    0,                // y
    nMaxDisplayWidth,    // cx
    rectDisplay.Height() +
    rectDisplay.top - rectDialog.top +
    abs (rectDisplay.left - rectDialog.left),    // cy
    SWP_SHOWWINDOW|SWP_NOMOVE);    // flags
}
}
```


BIBLIOGRAFÍA

Capítulo 1

- [1]<http://www.systemparking.com/company/history.htm> (Historia parking)
- [2]http://www.abc.es/hemeroteca/historico-16-02-2006/abc/Madrid/el-primer-parking-publico-de-espa%C3%B1a-en-la-plaza-de-santo-domingo-se-entierra-tras-50-a%C3%B1os_132312176254.html (Primer parking Madrid)
- [3]http://www.ing.uc.edu.ve/~emescobar/automat_l/contenido_menu/Unidad_l/Contenido/pagina2/pagina2.htm (Antecedentes de la automatización Industrial)

Capítulo 2

- [4]http://es.wikipedia.org/wiki/Unidad_de_control (Conceptos sobre Unidad de control)
- [5]<http://es.wikipedia.org/wiki/Sensor> (Conceptos básicos sobre sensores)
- [6]<http://www.controles.com/espanol/Automatismos/otros.htm> (Catalogo empresa de dedicada a la automatización industrial y telecomunicaciones)
- [7]http://www.exemys.com/site/pages/spanish/productos/otros_productos/IDx.htm (Empresa de dedicada al control industrial)
- [8]<http://www.siemens.es> (Link de la empresa Siemens)
- [9]http://www.accesor.com/esp/detail_product.php?id_article=31 (Catalogo empresa dedicada a los sistemas de seguridad y control de acceso)
- [10]http://www.accesor.com/esp/detail_product.php?id_article=77 (Catalogo empresa dedicada a los sistemas de seguridad y control de acceso)

- [11]http://www.accesor.com/esp/detail_product.php?id_article=88(Catalogo empresa dedicada a los sistemas de seguridad y control de acceso)
- [12]<http://es.wikipedia.org/wiki/ANPR>
- [13]<http://es.wikipedia.org/wiki/RS-485>
- [14]<http://es.wikipedia.org/wiki/RS-232>
- [15]<http://es.wikipedia.org/wiki/Profibus>
- [16]<http://es.wikipedia.org/wiki/Modbus>
- [17]<http://es.wikipedia.org/wiki/AS-interface>
- [18]<http://es.wikipedia.org/wiki/SCADA>
- [19]http://es.wikipedia.org/wiki/Base_de_datos

Capítulo 3

- [20]https://www.swe.siemens.com/spain/internet/webs/areas/productos/Pages/automatizacion_industrial.aspx
- [21]http://www.automation.siemens.com/simatic/controller/html_76/produkte/s7-300-standard-cpu.htm#CPU_315-2_DP (Servicio Siemens Automatización)
- [22]División PS. Productos y Sistemas industriales. SIEMENS S.A.
- [23]Máster ASP (Empresa de sistemas Parking)
- [24]Mabyc Levante S.L. (Empresa de sistemas Parking)
- [25] http://www.automation.siemens.com/cd/as-interface/index_76.htm (Características Bus AS-I)

Capítulo 4

- [26] http://www.ing.uc.edu.ve/~emescobar/automat_l/contenido_menu/Unidad_l/Contenido/pagina4/pagina4.htm (Pirámide CIM)
- [27] http://www.automation.siemens.com/salesmaterial-as/brochure/es/brochure_simatic-wincc-flexible_es.pdf (Generalidades WinCC)
- [28] http://es.wikipedia.org/wiki/Visual_Studio
- [29] <http://msdn.microsoft.com/es-es/library/ms167593.aspx> (Tutorial SQL Server)
- [30] <http://politecnica.uem.es/es/presentacion/instalaciones/arquitectura-de-computadoras-y-automatica> (Departamento de arquitectura de computadoras y automática de la UEM)

Capítulo 5

- [31] http://www.axis.com/products/cam_211/index.es.htm (Catálogo de cámara axis)
- [32] "Control del carril lineal y calibración de una cámara CCD en la adquisición 3D de piezas industriales"
- [33] Matrox Meteor II: installation and hardware reference
- [34] http://es.wikipedia.org/wiki/Computer_Vision (Aspectos generales de visión artificial)
- [35] http://www.matrox.com/imaging/products/intellicam/b_intellicam.pdf (Manual de la herramienta de trabajo Intellicam)